Herramientas del BIG DATA y analytics

Introducción al análisis de canastas de compra para *analytics translators* y científicos de datos (empleando R)

Autores:
Julio César Alonso
Ana María Arboleda

Universidad Icesi | Editorial Universidad Icesi

Introducción al Análisis de Canastas de Compra para analytics translators y científicos de datos (empleando R)

Julio César Alonso[1] Ana María Arboleda[2]

2025-06-06

[1]Universidad Icesi, Facultad de Negocios y Economía - Departamento de Economía, jcalonso@icesi.edu.co

[2]Universidad Icesi, Facultad de Negocios y Economía - Departamento de Mercadeo, Emprendimiento e Internacionalización, amarboleda@icesi.edu.co

© **Introducción al Análisis de Canastas de Compra para analytics translators y científicos de datos (empleando R)**

Julio César Alonso C. - Ana María Arboleda A.

Colección «Herramientas del Big Data y Analytics», vol. 5

Cali. Universidad Icesi, 2024.

132 páginas.

Incluye referencias bibliográficas.

ISBN: 978-628-7814-07-3 (eBook).

DOI: https://doi.org/10.18046/EUI/bda.h.7

Palabras Clave: 1. R | 2. Analítica | 3. Análisis de Canastas | 4. Minería de reglas | 5. Big Data Analytics

Clasificación Dewey: 545 ddc 21

CIENFI - Centro de Investigación en Economía y Finanzas

www.icesi.edu.co/centros-academicos/cienfi

Rector: Esteban Piedrahita Uribe

Secretaria General: Olga Patricia Ramírez Restrepo

Director Académico: José Hernando Bahamón

Coordinador editorial: Adolfo A. Abadía

Corrección de estilo: Claudia L. González G.

Diseño de portada: Sandra Moreno

Fotos tomadas por: Julio César Alonso

Editorial Universidad Icesi

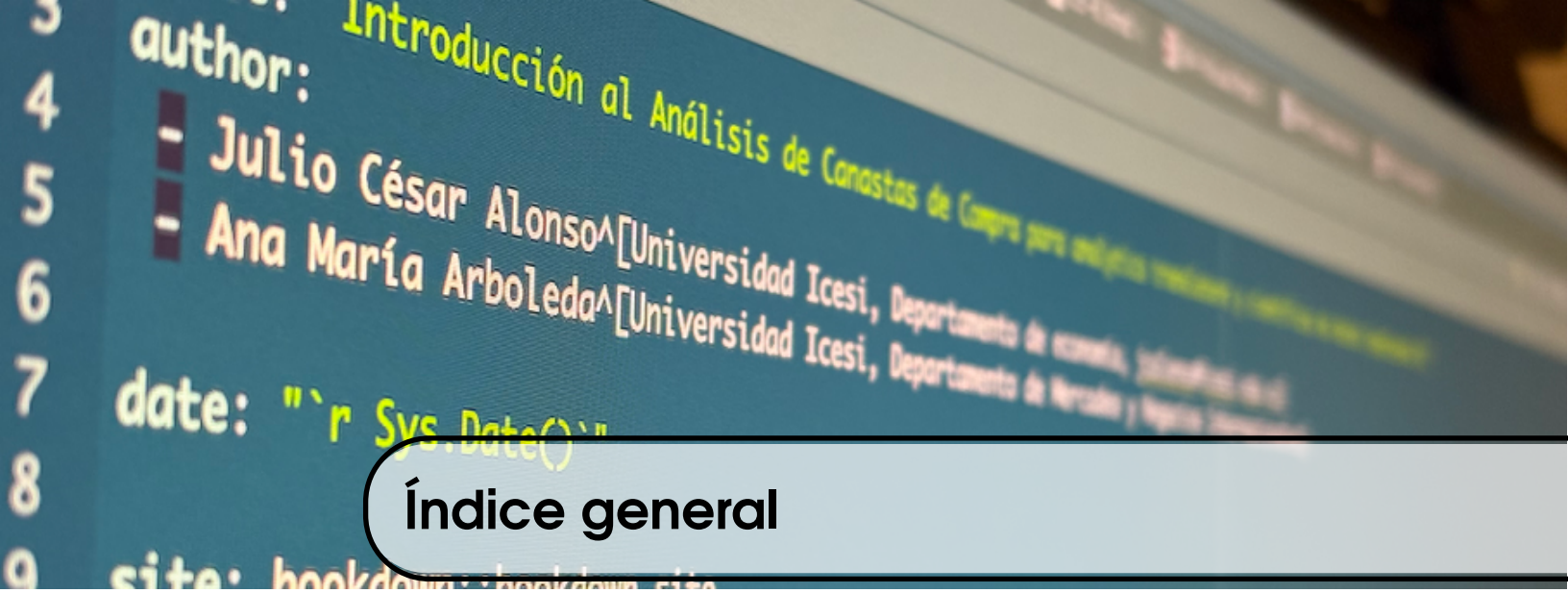
Calle 18 No. 122-135 (Pance), Cali – Colombia

Teléfono: +57 (2) 555 2334 | E-mail: editorial@icesi.edu.co

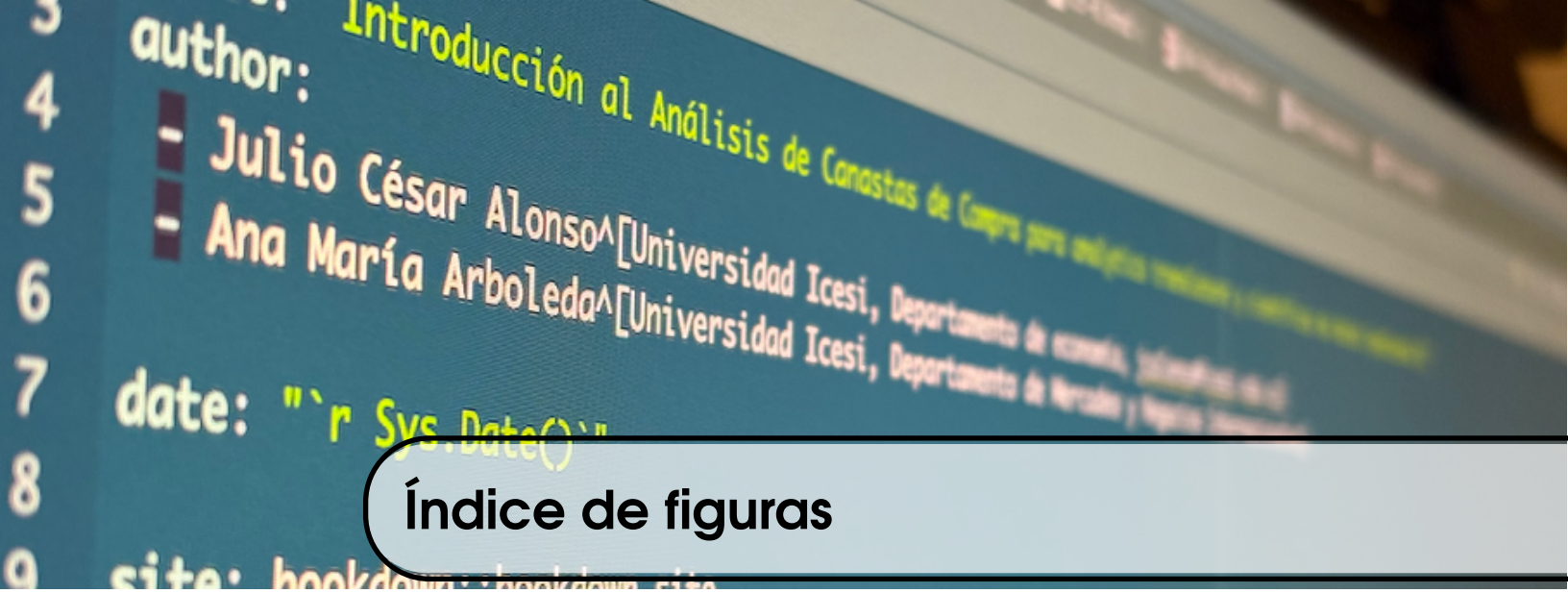
http://www.icesi.edu.co/editorial

Publicado en Colombia – *Published in Colombia*

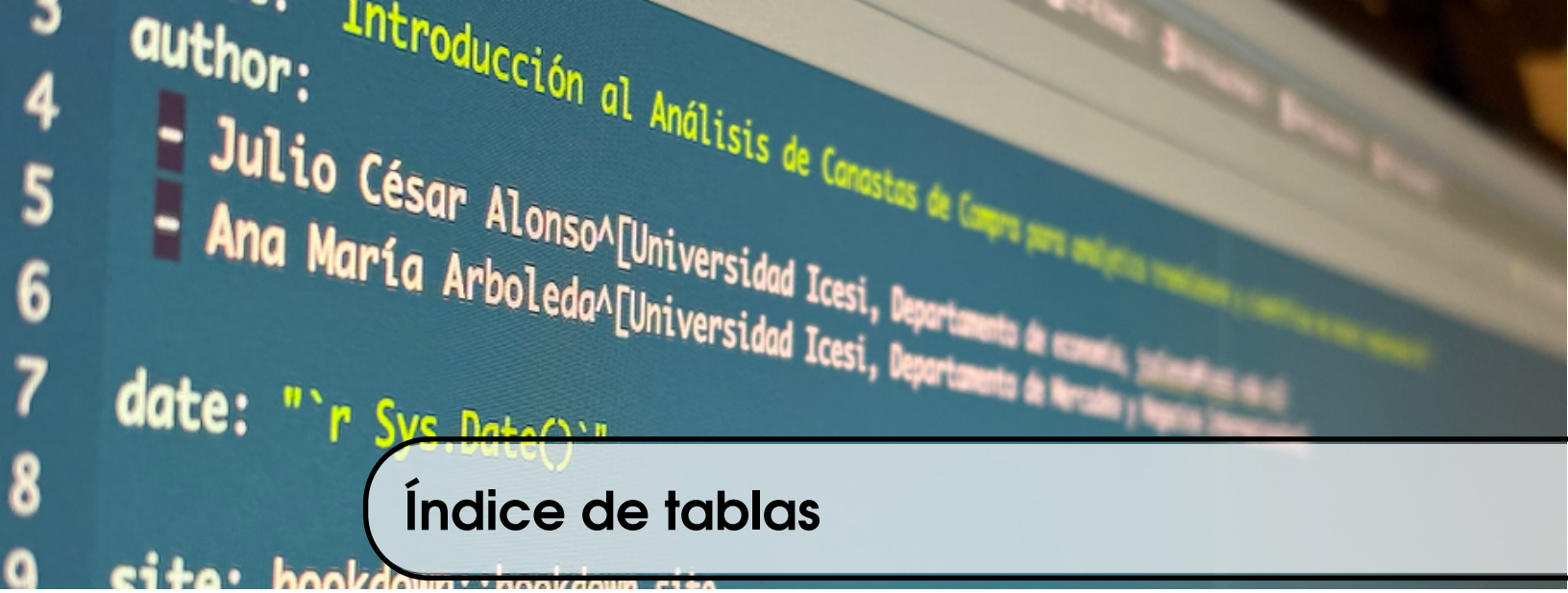
La publicación de este libro se aprobó luego de superar un proceso de evaluación doble ciego.



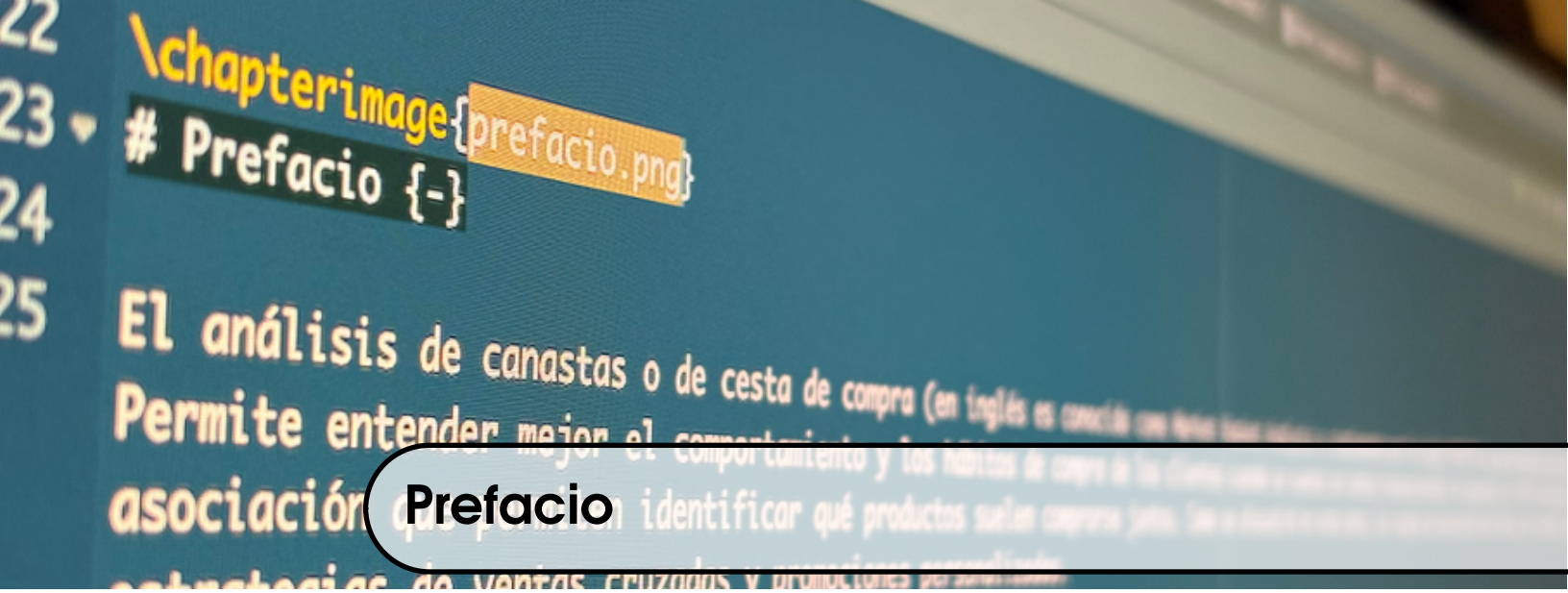
Índice general



Índice de figuras



Índice de tablas



Prefacio

El análisis de canastas o de cesta de compra (en inglés es conocido como Market Basket Analysis o simplemente por la sigla **MBA**) es una herramienta poderosa en el mercadeo. Permite entender mejor el comportamiento y los hábitos de compra de los clientes cuando se cuenta con datos transaccionales. En especial, el **MBA** encuentra reglas de asociación que permiten identificar qué productos suelen comprarse juntos. Como se discutirá en esta obra, las reglas de asociación son útiles, por ejemplo, para desarrollar estrategias de ventas cruzadas y promociones personalizadas.

Este libro está dirigido a dos roles en el mundo del *business analytics*: el científico de datos y el *analytics translator*. Normalmente, detrás del proceso de transformar datos en *insights* para la toma de decisiones existe un equipo con profesionales con diferentes roles y competencias. En estos equipos son pieza clave el científico de datos y el *analytics translator*. El científico de datos estima y entrena modelos estadísticos y de inteligencia artificial para resolver las preguntas de negocio planteadas.

El *analytics translator* facilita la comunicación entre el equipo de científicos de datos, los tomadores de decisiones, así como con los *stakeholders* del proyecto. Su rol principal es traducir el lenguaje técnico del científico de datos a un lenguaje comprensible para los tomadores de decisiones. Además, el *analytics translator* se encarga de identificar las necesidades y objetivos del negocio, para asegurarse de que el trabajo del científico de datos esté alineado con ellos. De esta manera, el *analytics translator* juega un papel fundamental en el éxito de un proyecto de análisis de datos, garantizando una comunicación clara y efectiva entre el equipo técnico realizando los cálculos y las personas tomando las decisiones.

En este orden de ideas, el Capítulo 1 discute qué es el **MBA**, para qué es útil y cómo encaja este tipo de análisis en las tareas y tipos de analítica. Así mismo, este capítulo presenta una historia mítica que da origen a este campo de estudio. El Capítulo 1 está escrito tanto para científicos de datos como para *analytics translators*. En este se discuten conceptos importantes de mercadeo que emplean los tomadores de decisiones; conceptos que deberían conocer los científicos de datos.

El Capítulo 2 presenta los conceptos que se emplean en el **MBA**, como regla de asociación, las métricas y el algoritmo *Apriori*. Para lograr esto, a lo largo del capítulo, se emplea un ejemplo sencillo que termina con una discusión de cómo se pueden emplear los resultados para la toma de decisiones. Este capítulo le permitirá a científicos de datos y *analytics translators* establecer un lenguaje común y una comprensión del **MBA** que les facilitará establecer una comunicación fluida.

El Capítulo 3 desarrolla un ejemplo de análisis de canasta de inicio a fin empleando una base de datos extensa de una empresa de comercio electrónico con sede en el Reino Unido. Se muestra cómo emplear R para llevar a cabo el análisis que se presentó en el Capítulo 2. Si bien este capítulo está escrito principalmente para científicos de datos que emplean R, puede ser útil para el rol de *analytics translators* (sin necesidad de todo el detalle técnico) los pasos que deben realizar los científicos de datos para obtener reglas de decisión a partir del **MBA**.

El entender la creación de las reglas de asociación le permitirá al *analytics translator* tener una mayor comprensión del tiempo y el trabajo involucrado en un análisis de canasta. Como parte de un equipo, es importante que cada uno de los miembros tenga un conocimiento general de los conceptos (esto lo logra el Capítulo 2) y procesos involucrados en el **MBA** (esto lo logra el Capítulo 3). El conocimiento mutuo de los roles facilitará la colaboración y la comprensión dentro del equipo.

El Capítulo 4 muestra diferentes alternativas para visualizar y presentar los resultados de un **MBA**. El científico de datos construirá las visualizaciones y el *analytics translator* las empleará para comunicar los resultados a las personas que toman decisiones. Así mismo, las tablas con las reglas de decisión serán una herramienta importante para los tomadores de decisiones. De esta manera, tanto el científico de datos como el *analytics translator* deberán acordar la mejor forma de presentar y visualizar los resultados del **MBA** para garantizar una comunicación clara y efectiva. Este capítulo le permite a ambos roles conocer qué tipo de visualización y presentación de resultados están disponibles, a la vez que le muestra a los científicos de datos cómo construir dichas visualizaciones.

Finalmente, el Capítulo 5 presenta un segundo ejemplo más elaborado de un **MBA** empleando datos de una panadería en línea de Edimburgo. Este ejemplo le permitirá tanto al rol de científicos de datos como al de *analytics translator* explorar con mayor profundidad cómo el **MBA** puede ser empleado sobre diferentes segmentos de los consumidores para brindar insights al negocio que lleven a decisiones accionables.

El libro está construido de tal manera que sea útil tanto para el rol de científico de datos como para el de *analytics translator*. Los lectores con el rol de *analytics translator* podrán saltar el detalle técnico del código en R en los capítulos 3 y 4 sin miedo a perder la continuidad del texto.

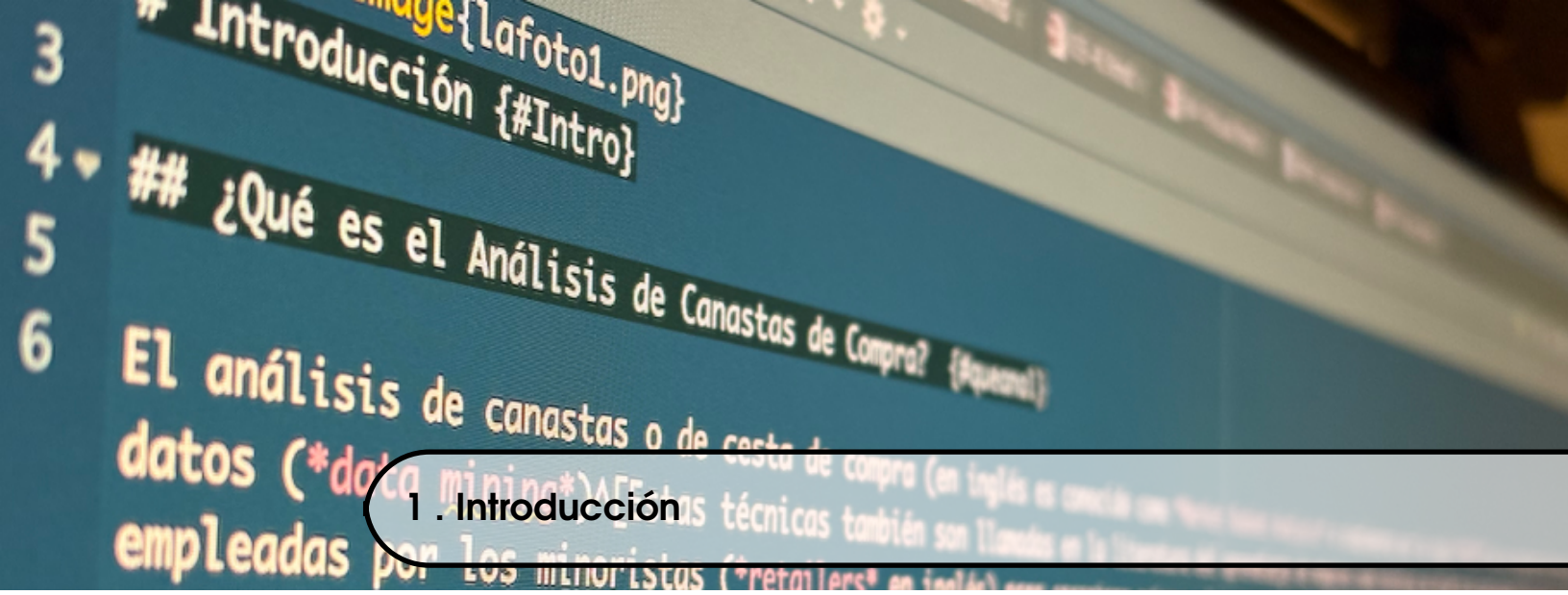
Para los lectores en el rol de científico de datos, es importante mencionar que este libro supone un uso intermedio de R. Si crees que necesitas algún refuerzo en R, recomendamos tres libros. Alonso y Ocampo (2022) presenta una breve introducción para iniciar a usar R. Ese primer libro discute cómo instalar R y RStudio y paquetes, cómo cargar diferentes bases de datos y cómo realizar operaciones aritméticas y lógicas con

objetos. En Alonso y Ocampo (2022) también se discuten las clases esenciales de objetos sencillos y compuestos. No dudes en consultar ese primer libro si aún no has iniciado tu camino por el universo de R.

El segundo libro de la serie (Alonso, 2022), presenta una breve introducción al paquete para *dplyr* (Wickham et al., 2021) que permite manipular objetos que contengan datos. En ese libro se discute cómo filtrar observaciones, crear nuevas variables y combinar objetos con datos. Es recomendable tener un conocimiento de ese paquete antes de leer esta obra. Consulta ese segundo libro si aún no has tenido alguna experiencia manipulando objetos con datos con *dplyr*.

Finalmente, recomendamos (Alonso y Largo, 2023), en el que se presenta una introducción a la creación de visualizaciones con el paquete *ggplot2* (Wickham, 2016). En esta obra trabajaremos en visualizaciones empleando el paquete *ggplot2*. Así, este libro asume un manejo intermedio de R y de los paquetes *dplyr* y *ggplot2*.

Esta obra recoge nuestra experiencia trabajando con R y el **MBA** para la transformación de datos en conclusiones que faciliten la toma de decisiones en organizaciones privadas y públicas. **¡Esperamos encuentres esta obra útil y la compartas con otros!** Si tienes alguna sugerencia del libro o corrección, no dudes en escribirnos. Esta es una obra en constante construcción.



1 . Introducción

1.1 ¿Qué es el Análisis de Canastas de Compra?

El análisis de canastas o de cesta de compra (en inglés es conocido como *Market Basket Analysis* o simplemente por la sigla **MBA**) es un conjunto de técnicas de minería de datos (*data mining*)[1] empleadas por los minoristas (*retailers* en inglés) para encontrar patrones de compra de sus clientes con el fin de aumentar las ventas.

El **MBA** es un tipo de aprendizaje de máquina no supervisado[2] que emplea algoritmos y datos de transacciones (tirillas de compra) para encontrar combinaciones de artículos que aparecen juntos con frecuencia en las transacciones; es decir, que tienen co-ocurrencia. Estos algoritmos tienen como finalidad encontrar "reglas de asociación" que permitan determinar con una alta probabilidad qué producto (ítem) será comprado dado que ya se tienen en la canasta un determinado conjunto de ítems.

El **MBA** permite encontrar qué artículos compran frecuentemente los clientes juntos, generando un conjunto de **Reglas de Asociación**. Es decir, el **MBA** genera reglas de la forma "si pasa esto, entonces ocurre aquello" (en la jerga del *business analytics* este tipo de reglas se conocen con la sigla **IFTTT** que viene del inglés *if this then that*). En nuestro contexto, las reglas pueden ser del tipo "si ya se tiene en la canasta el producto A, entonces es probable que se incluyan los productos B y C en la canasta". Estas reglas pueden ser empleadas para tomar decisiones de mercadeo, que responden a preguntas de negocio como ls que se discuten a continuación.

¿Qué producto "impulsa" la compra de otros productos? Aunque las personas tengan en su mano una lista de compras, los productos no son para el consumidor un

[1] Estas técnicas también son llamadas en la literatura del aprendizaje de máquina como técnicas de minería de asociación (*Association Mining*).

[2] En el campo del aprendizaje de máquina se distinguen dos tipos de aprendizaje: supervisado y no supervisado. El aprendizaje supervisado se caracteriza por emplear datos de entrenamiento etiquetados; es decir, con la respuesta correcta ya conocida. Por otro lado, en el aprendizaje no supervisado los datos no están etiquetados. Es decir, no tiene datos que contengan la "respuesta correcta". El aprendizaje no supervisado busca descubrir patrones o estructuras ocultas en los datos.

listado independiente. Los productos se entienden bajo agrupaciones conceptuales o categorías[3] (*categorización*).

Por ejemplo, en la categoría de aseo personal encontramos jabón para el cuerpo o manos, shampoo, acondicionador, desodorante, cepillos de dientes. La subcategoría de shampoo se puede clasificar por tipo de cabello: rubio, negro, rizado, seco, graso. En este ejemplo, supongamos que el **MBA** muestra que cuando las personas compran acondicionador (producto A), es altamente probable que compren shampoo (producto B). De esta manera, tiene sentido crear una oferta *shampoo + acondicionador* para recordar y motivar la compra del producto que tiene baja rotación. Así, el **MBA** facilita estrategias exitosas en la administración de la categoría o *category management*[4] (*categorización*).

¿Cómo debo exhibir los productos en el punto de venta de forma diferente a la convencional? El punto de venta es el lugar en el que el consumidor puede encontrar y comprar el producto; este concepto se conoce como canal de venta[5] (*canales de marketing*).

El **MBA** puede mostrar cómo enriquecer creativamente una categoría de productos, lo cual agregaría valor al consumidor. Por ejemplo, las frutas y verduras son una gran categoría y están todas ubicadas en la misma área (además por condiciones de refrigeración). Supongamos que el **MBA** muestra que las personas que compran pepino (producto A) también compran con una alta probabilidad *lemon pepper*[6] (producto B). Una buena proporción de consumidores que compran pepino tal vez no había pensado en esta combinación. De esta manera sería pertinente crear una exhibición cruzada ubicando el *lemon pepper* junto al pepino. La exhibición cruzada es una forma de marketing cruzado[7] (*cross merchandising*); en este caso, el cruce de

[3]En mercadeo se usa el concepto de categorización para construir grupos de productos en la mente del consumidor (Rooderkerk y Lehmann, 2021). Esto es útil porque facilita la toma de decisiones al consumidor (Arboleda y Arce-Lopera, 2015). Al clasificar un objeto bajo una categoría, le hace más fácil a la persona recordarlo y rápidamente inferir atributos y beneficios de acuerdo con los demás elementos de la categoría. En el canal minorista, la construcción de categorías permite organizar las exhibiciones de acuerdo con la forma como se agrupan conceptualmente los productos. En el punto de venta, las categorías permiten organizar los artículos en la góndola de la tienda mostrando al consumidor todas las opciones disponibles que permiten responder a una necesidad que es coherente con la función o beneficio que ofrecen.

[4]*Category Management* es la ubicación de productos en el punto de venta de tal forma que faciliten al consumidor la toma de decisión y a la tienda la venta de productos (Pascucci et al., 2022). A través del *category management* se debe entender cuál producto debe estar junto a otro para lograr una mayor venta. Esta ubicación de los productos en la categoría puede además estar asociada a un descuento o paquete promocional.

[5]La razón de ser de los canales de marketing como tiendas, supermercados u otros puntos de venta (POP o *point of purchase*) es crear un "espacio" donde el consumidor pueda encontrar el producto. El canal es rentable en la medida en que este espacio que se ocupa con un producto se desocupa rápidamente. Esto se conoce como rotación del inventario. Si el producto no rota rápidamente, o no se vende, la tienda no va a ser rentable y debe cerrar su oferta. El valor que tiene el punto de venta para el consumidor es permitirle encontrar los productos que necesita y para esto el canal debe entender cómo exhibir los productos.

[6]El *lemon pepper* es un condimento cuya traducción sería pimienta con limón.

[7]El marketing cruzado, *cross merchandising* o promoción cruzada es una estrategia de marketing visual para el punto de venta en la que las empresas o marcas se unen para ayudarse mutuamente a aumentar las ventas. La exhibición cruzada permite "sacar" a un producto de su categoría y llevarlo a otra en la que no corresponde, pero su presencia permite entender el producto de la exhibición original de una forma diferente. De esta manera, las exhibiciones cruzadas se pueden utilizar para recordar o motivar al consumidor

un producto a la otra categoría sugiere visualmente a las personas que esta es una buena combinación y motiva la compra de ambos productos (Drèze y Hoch, 1998). Otra opción es que un promotor de ventas haga la recomendación; dado que la persona quiere comprar A, le recomienda comprar B. Esto se conoce como venta cruzada (Coss-selling) e implica que ya el consumidor quiere comprar un producto inicial y un vendedor o un sistema de recomendación automatizado le sugiere un producto complementario (Kocas et al., 2018).

¿Puedo sacar de mi portafolio este producto sin afectar las compras de otros productos? Para el consumidor, el canal de distribución o punto de venta debe generar valor, permitirle acceder a productos que se ajusten a sus necesidades. Por ejemplo, la razón de ser de una librería es vender libros, aunque cada vez las personas leen menos libros o incluso sustituyen el libro físico por un e-book. El café no es la razón de ser de una librería. Entonces, ¿debería eliminar el café de su portafolio de productos y dedicarse a vender libros?

A primera vista, sería mejor concentrarse en los libros y no "molestarse" con el inventario y producción de café. Sin embargo, el **MBA** nos puede mostrar en qué medida la compra de libros realmente ocurre simultáneamente a la compra de café; los libros coocurren con el café. Entonces, aunque no sea el producto principal, es el "ancla" que lleva a las personas a la librería; buscan un café y, ya que están allí, llevan un libro que se antojan de leer en ese momento. Este ejemplo tiene además el objetivo de señalar que el **MBA** permitiría identificar estrategias para un negocio que debe transformarse.

El **MBA** nos muestra qué artículos están asociados a comprar un libro. De esta manera, el tomador de decisiones debe interpretar estrategicamente los resultados del **MBA** y formular preguntas pertinentes que se puedan responder con dichos resultados. Por ejemplo: ¿qué artículos están comprando las personas que me permiten entender su motivación por entrar a la tienda (de libros)? Y una pregunta más abstracta: ¿qué es lo que la persona está buscando en la librería? Se podría llegar a conclusiones tan inesperadas como que la librería no "vende libros", vende un espacio de tranquilidad y relajación; al ofrecer un café (y no un libro), la frecuencia de visita y de compra puede ser mayor.

¿Qué productos tienen sentido poner en un "combo" promocional? El combo une productos que son complementarios o conceptualmente coherentes[8] (*combo*). Por ejemplo, supongamos que una tienda vende solo cuadernos, lápices, crispetas y gaseosas. ¿Cuáles productos son complementarios? Es decir, ¿cuáles productos podrían unirse para formar un combo?

Convencionalmente, la construcción de combos se realiza de manera intuitiva. Sin embargo, también se puede basar en datos obtenidos a través de los resultados del **MBA**. Los combos tienen sentido para el fabricante y el canal de distribución porque motivan la venta de ambos productos, o quizás impulsen la venta de un producto que no tiene alta rotación. El combo tiene sentido para el consumidor porque reduce el

a llevar un producto que no estaría comprando habitualmente, pero que puede querer como un buen complemento.

[8] Un combo es un conjunto de productos que se ofrecen como un solo paquete a un precio menor que si el consumidor decidiera comprarlos por separado.

precio total de ambos productos y porque le permite recordar que hay un producto que quizás va a necesitar en una ocasión cercana al consumo del primero.

Por ejemplo, la persona tiene una herida y va rápidamente a la farmacia para comprar un medicamento que le permita desinfectar la herida. En el afán, puede olvidar que necesita algo para cubrir la herida (una gaza). Un combo puede ofrecer el desinfectante de herida con una gaza estéril. Por otro lado, el **MBA** también nos puede mostrar qué productos son complementarios y casi siempre se llevan juntos, de tal manera que crear un combo no es necesario para estimular su consumo y se pueden vender a su precio regular sin necesidad de generar un descuento.

¿Qué cliente que no está consumiendo un producto tiene una alta probabilidad de comprarlo? Las decisiones de compra[9] de consumidores habituales permite encontrar patrones que pueden ser generalizables a otros consumidores con ciertas características (en este caso canastas), aunque estos consumidores no sean conscientes de ese patrón. Por ejemplo, una persona que inicia en la afición de los asados compra todo lo que piensa que puede necesitar: la parrilla (gas o carbón), carne, cuchillos y tabla para cortar. Sin embargo, la persona no es consiente de la necesidad de un utensilio para girar la carne en la parrilla, buscando la cocción ideal.

La regla de asociación fruto del **MBA** puede sugerir que compre unas pinzas (producto B), dado que ya tiene en su cesta la parrilla (gas o carbón), carne, cuchillos y tabla para cortar (conjunto de productos A). El **MBA** no solo permite motivar la compra de productos coherentes dentro de un sistema, sino que es de considerable ayuda para el consumidor al hacerle recomendaciones que le faciliten la vida. Por ejemplo, para la celebración de cumpleaños de un ser querido, compras la torta, el helado, incluso la decoración que dice "feliz cumpleaños". Esta celebración es crucial, pero solo ocurre una vez al año y quizás no eres experto organizador de fiestas de cumpleaños. Entonces, cuando llegas a casa, te das cuenta de que olvidaste las velas para la torta. El **MBA** no solo permite hacer un sistema de recomendaciones, sino también listas de compra para diferentes ocasiones, generando valor a los consumidores y mayor fidelidad.

¿Será necesario el descuento en este producto? La táctica de descuento en el precio de un producto[10] (*descuento*) puede utilizar los resultados del **MBA** para determinar su pertinencia. A partir del **MBA**, es posible determinar qué producto A puede tener en descuento para motivar la compra del producto B. Es decir, al disminuir el precio de un

[9] Los consumidores toman decisiones de acuerdo a sus hábitos, experiencia, según lo que conocen y lo que antes han hecho. En la medida en que el consumidor aprende de un producto y tiene más experiencia con una situación de consumo, puede cambiar o modificar los productos que necesita y demanda.

[10] El descuento o la reducción de precio busca que un producto se venda más rápido o llegue a un mayor número de personas. La decisión de descuento debe ser estratégica al considerar las implicaciones que tiene en la confianza por el producto y el aumento en la demanda (Arboleda y Alonso, 2016). Si el producto baja de precio, sin un argumento temporal, el consumidor podrá tener la percepción de que el producto ha disminuido su calidad o cantidad. Por el contrario, al argumentar que hay un descuento "limitado" en el tiempo o en unidades, el consumidor podrá entender que el producto no cambia sus atributos y beneficios, pero podrá obtenerlo a un menor precio por tiempo limitado. En este caso, el tendero podrá esperar un aumento en la demanda del producto. La idea de tiempo limitado genera en el consumidor la percepción de escasez y, bajo esta idea, motiva la compra. Sin embargo, esta "escasez" debe ser real o temporal; de lo contrario, genera la idea de engaño.

producto, aumentaría la compra de dicho producto y, como consecuencia, es posible afectar la compra de otros productos que no están en descuento.

Por ejemplo, una tienda de artículos para el colegio ofrece los morrales con un 30% de descuento. El **MBA** me muestra que, al comprar un morral, las personas compran lonchera. Entonces el descuento en morrales también impulsa la compra de loncheras. De esta manera, el **MBA** me permite saber sobre qué producto debo hacer el descuento. Al aplicar el descuento sobre la lonchera, el consumidor no va a comprar muchas más loncheras, y el descuento tampoco impulsa la compra del morral. Es importante anotar que el **MBA** no nos permitirá decidir sobre el monto del descuento.

¿Puedo hacer una oferta personalizada? Las estrategias de marketing (Kotler y Armstrong, 2012) utilizan cada vez menos el marketing masivo[11] (*marketing masivo*) y en cambio priorizan necesidades del consumidor bajo la idea de marketing de nicho[12] (*marketing de nicho*) o marketing personalizado[13] (*marketing personalizado*). A través del **MBA** es posible emplear patrones de compra de productos de otros consumidores para personalizar la experiencia de otro consumidor con una canasta similar y así realizar una comunicación o descuento coherente con sus hábitos.

Por ejemplo, en la categoría dermocosmética, una persona de 15 años compra un producto para el acné. El **MBA** puede además recomendarle el limpiador facial propio para su condición y además recordarle, cuando se está agotando el producto, continuar su tratamiento y complementarlo con productos seguros para su necesidad específica, dado que esos son los productos (conjunto de productos B) que típicamente lleva en su canasta un consumidor que ya tiene un producto para el acné (producto A).

¿Qué tener en cuenta al diseñar piezas de comunicación virtuales o físicas? Al entender que la compra de un producto A se asocia a la compra del producto B o C, quienes diseñan piezas publicitarias obtienen información valiosa para inspirar su creatividad de los resultados de un **MBA**. La comunicación puede hacer énfasis en un descuento por combo de productos (A+B). Descuento en vinagreta por la compra de canasta de verduras. También puede tener un diseño que considere el contexto de consumo. De esta manera, la comunicación sugiere que, al comprar el producto A, recuerde llevar el producto B. Estos combos también pueden ser alianzas entre diferentes marcas o empresas (Arboleda y Alonso, 2016). Por ejemplo, si invita a los amigos el fin de semana a ver el partido, el anfitrión puede ofrecer perros calientes (producto

[11] El marketing masivo realiza una oferta de producto, estrategias de comunicación, canal y precio que lleguen a "todas" las personas sin establecer prioridades por segmentos de consumidores.

[12] El marketing de nicho realiza una segmentación del consumidor de manera específica entendiendo su ubicación geográfica, condiciones demográficas, características propias de su momento de vida, motivaciones y hábitos de consumo. La especificidad del grupo objetivo, señala un grupo de personas relativamente reducido en comparación con el marketing masivo. De acuerdo con la definición de preferencias del segmento-nicho, se diseña un producto para este grupo de consumidores específicamente (target) y de manera coherente se toman decisiones de precio, canal y comunicación.

[13] El marketing personalizado establece con el consumidor una relación uno a uno, busca apelar a intereses y necesidades específicas. Con este objetivo, el marketing personalizado requiere de herramientas tecnológicas que puedan "entender" al consumidor en su historia de consumo del producto, otros productos, u comportamientos observados sistemáticamente para así hacer recomendaciones específicas al interés de cada persona.

A) acompañados por cerveza (producto B) y papitas (producto C). El **MBA** permite guiar la creatividad en la comunicación al ofrecer productos complementarios en la canasta de compra.

El **MBA** permite responder este tipo de preguntas estratégicas para el negocio al observar una relación entre un producto o un conjunto de ellos y otros productos con los que se asocia. En últimas, se espera que el **MBA** pueda aumentar las ventas y la satisfacción de los clientes.

Empleando los datos de las canastas de compra (transacciones) para determinar qué productos se compran frecuentemente juntos, los minoristas pueden optimizar la colocación de los productos, realizar ofertas especiales y crear nuevos paquetes de productos para fomentar más ventas de estas combinaciones.

El objetivo último del **MBA** es que los minoristas puedan tomar acciones, basadas en datos, que generen ventas adicionales, al tiempo que la experiencia de compra sea más personalizada y valiosa para los clientes. Al utilizar el **MBA**, se puede generar una mejor experiencia de compra a los clientes y generar un mayor sentimiento o lealtad de marca hacia la organización.

1.2 Una historia mítica

Antes de entrar en el detalle, es importante contar una historia que se ha convertido en mito sobre el **MBA**. Es común escuchar una historia sobre **cerveza y pañales** que se cuenta en los salones de clase cuando se realiza la introducción al tema del **MBA**. ¡Y este libro no puede ser la excepción! La historia es más o menos la siguiente.

En una cadena de supermercados en los Estados Unidos, a inicios de la década de los 90, unos analistas empezaron a combinar las diferentes fuentes de datos disponibles para entender mejor los patrones de compra en cada uno de los almacenes. Los analistas combinaron los datos de su sistema de tarjetas de fidelidad con los de sus sistemas de los puntos de venta.

La primera base de datos proporcionaba datos demográficos de los clientes y la segunda base de datos contenía las tirillas de compra de sus clientes. Es decir, la segunda base les permitía conocer dónde, cuándo y qué compraban sus clientes. Tras combinar los datos, se encontraron patrones reveladores, pero otros obvios. Por ejemplo, por el lado de las reglas obvias, se encontró que los clientes que compran ginebra también suelen comprar agua tónica y a menudo también compran limones (para hacer *gin and tonic*).

Pero no todo era obvio. Los analistas descubrieron que cuando los hombres compraban pañales los jueves y los sábados, también tendían a incluir en su carrito de compra cerveza. Un análisis más detallado demostró que estos compradores solían hacer sus compras para toda la semana los sábados y los jueves solo compraban algunos artículos. Los analistas concluyeron que los hombres jóvenes compraban la cerveza para tenerla disponible para el fin de semana.

La historia continúa con que la cadena decidió acercar la exhibición de la cerveza

junto a la de los pañales. Además, no ofreció ningún tipo de descuento o combo para la cerveza y los pañales los días jueves. Este es el conocido "mito" de la cerveza y los pañales. Todo parece ser un mito, pues no está documentado que esto en efecto haya ocurrido. Algunas personas narran la historia diciendo que la cadena de almacenes era del medio oeste (*Midwest*) de los Estados Unidos. Otras versiones de la historia dicen que Walmart es esa cadena de supermercados. Pero esta versión de la historia ha sido desmentida por fuentes de Walmart (ver, por ejemplo, Contemporary Analysis (2022) o Power (2002)).

Realmente, todo parece más un mito que permite explicar el potencial de las técnicas de **MBA**. Pero, como todo mito, esta historia parece que está basada en algo de realidad. Diferentes fuentes (como por ejemplo Madsen (2017), Power (2002), Swoyer (2016) o CBR (1998)) cuentan que en 1992 un grupo de consultores de la compañía Teradata analizaron aproximadamente 1,2 millones de canastas de mercado (tirillas de compra) de aproximadamente 25 almacenes de la cadena Osco Drug. El análisis permitió concluir que entre las 5 y las 7 de la tarde los consumidores compraban cerveza y pañales. También está claro que en ese momento los gerentes de Osco no emplearon esta relación para acercar los lugares de exhibición de los dos productos, como dice el mito.

Este es el mito de la cerveza y los pañales que inspira a todos los estudiantes de esta área del *business analytics*. Así como en el mito, esperamos que este libro te ayude a encontrar canastas inesperadas que potencialicen el valor de la compra que hace el consumidor y que fortalezcan al negocio minorista.

1.3 El MBA y el Business Analytics

Ya sea un mito o no la historia de la cerveza y los pañales, en todo caso lo que sí es un hecho es que el *Market Basket Analysis* (**MBA**) se emplea en la actualidad para identificar patrones de compra y mejorar la estrategia de ventas. Hoy las técnicas para realizar **MBA** hacen parte de la caja de herramientas del *business analytics*. **El *business analytics* es el proceso científico de transformar datos en *insights* con el propósito de tomar mejores decisiones** (Ver Capítulo 1 de Alonso (2024) para una mayor discusión).

El **MBA** es una tarea del *business analytics* que emplea algoritmos para descubrir qué productos suelen comprarse juntos y así ofrecer recomendaciones personalizadas a los clientes. En el proceso científico de transformar datos en *insights* existen diferentes actividades que van desde la recolección de datos y su almacenamiento hasta la toma de la decisión, pasando por la extracción, limpieza y preparación de los datos, su exploración y visualización y el modelado que se requiera para responder a la pregunta de negocio planteada.

Como se discute en el Prefacio, este libro está dirigido tanto a científicos de datos como a *analytics translators*[14]. El científico de datos es la persona que organiza los

[14]Las actividades que permiten pasar de datos a *insights* para la toma de decisiones son desarrolladas por un equipo con profesionales calificados que tienen diferentes competencias y roles. En estos equipos son pieza clave el científico de datos y el *analytics translator*. El científico de datos estima y entrena modelos estadísticos y de inteligencia artificial para resolver las preguntas de negocio planteadas. El *analytics trans-*

Figura 1.1. Material multimedia: roles en la analítica

Fuente: elaboración propia. https://youtu.be/rhLWa-vOxyU

datos y descubre las reglas de asociación del **MBA**. El *analytics translator* es quien tiene en mente la estrategia del negocio, y de acuerdo con las decisiones que desea tomar la organización, establece prioridades en las reglas de asociación para finalmente llevar los resultados del **MBA** a la práctica. Aunque son roles diferentes, ambos deben estar coordinados para generar reglas de asociación y decisiones pertinentes para el negocio.

Por otro lado, al centrarnos en lo que se puede hacer con los datos en el mundo del *business analytics*, podemos clasificar las actividades realizadas con los datos en diferentes tareas. Como se discute en Alonso (2024), las tareas que se desarrollan en el *business analytics*, en la mayoría de los casos, se pueden clasificar en ocho[15]:

1. Clasificar
2. Hacer regresiones
3. Detectar anomalías
4. Formar clústeres
5. Hacer pronósticos
6. Visualizar
7. Resumir datos

lator facilita la comunicación entre el equipo de científicos de datos, los tomadores de decisiones, así como con los *stakeholders* del proyecto. Su rol principal es traducir el lenguaje técnico del científico de datos a un lenguaje comprensible para los tomadores de decisiones. Además, el *analytics translator* también se encarga de identificar las necesidades y objetivos del negocio, para asegurarse de que el trabajo del científico de datos esté alineado con ellos. De esta manera, el *analytics translator* juega un papel fundamental en el éxito de un proyecto de análisis de datos, garantizando una comunicación clara y efectiva entre todas.

[15]Ver Capítulo 1 de Alonso (2024) para una descripción de cada una de estas tareas.

8. **Encontrar Reglas de Asociación**

Por otro lado, es común clasificar los ejercicios de analítica en cuatro tipos de analítica según el propósito del análisis:

- **Analítica Descriptiva**: Esta analítica se enfoca en resumir y visualizar los datos para obtener información sobre lo que ha sucedido en el pasado. Ayuda a comprender patrones y tendencias.
- **Analítica Diagnóstica**: Esta analítica busca entender por qué algo ha sucedido. Examina los datos para identificar las causas raíz de los problemas o éxitos pasados.
- **Analítica Predictiva**: Esta analítica utiliza modelos estadísticos y de aprendizaje de máquina para hacer pronósticos y predecir eventos futuros.
- **Analítica Prescriptiva**: Esta analítica se centra en recomendar acciones y soluciones óptimas para lograr un objetivo.

Figura 1.2. Material multimedia: tipos de analítica

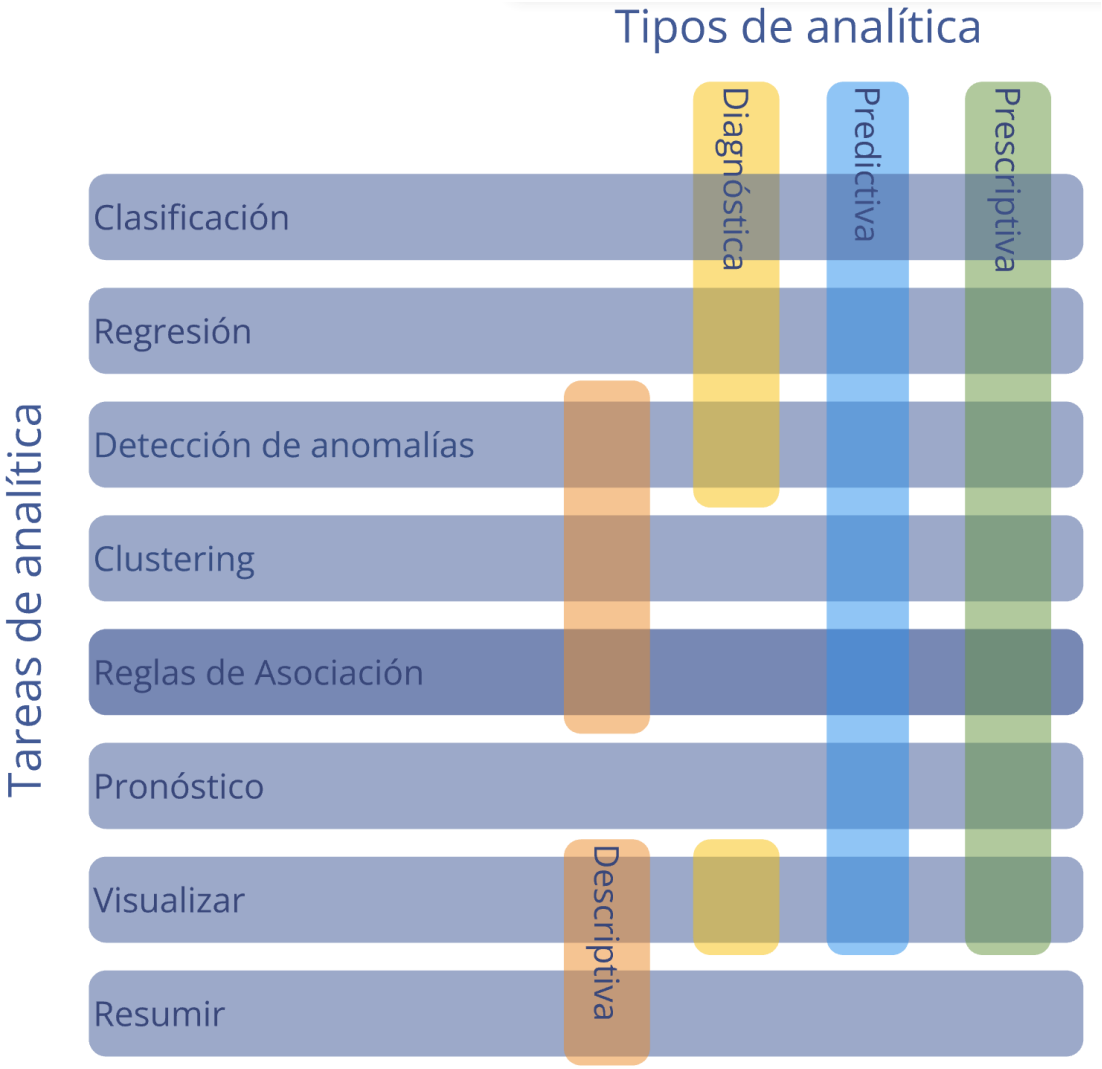
Fuente: elaboración propia. https://rb.gy/s7efwr

Estos cuatro tipos de analítica engloban las 8 tareas del *business analytics*. El **MBA** se emplea para realizar la **tarea de encontrar asociaciones** que implica realizar analítica descriptiva y prescriptiva (Ver Figura 1.3).

La **analítica descriptiva** responde a la pregunta: ¿qué está pasando en mi negocio? El **MBA** puede ayudar a describir cómo es la composición de las canastas que compran los consumidores y cuáles son más frecuentes que otras. Esto lo estudiaremos en el Capítulo 2.

La **analítica predictiva** permite responder a la pregunta: ¿qué ocurrirá en mi negocio? El **MBA** permite hacer conjeturas de que otros productos se incluirán en la canasta

Figura 1.3. Relación entre las tareas de analítica y los tipos de analítica



Fuente: elaboración propia.

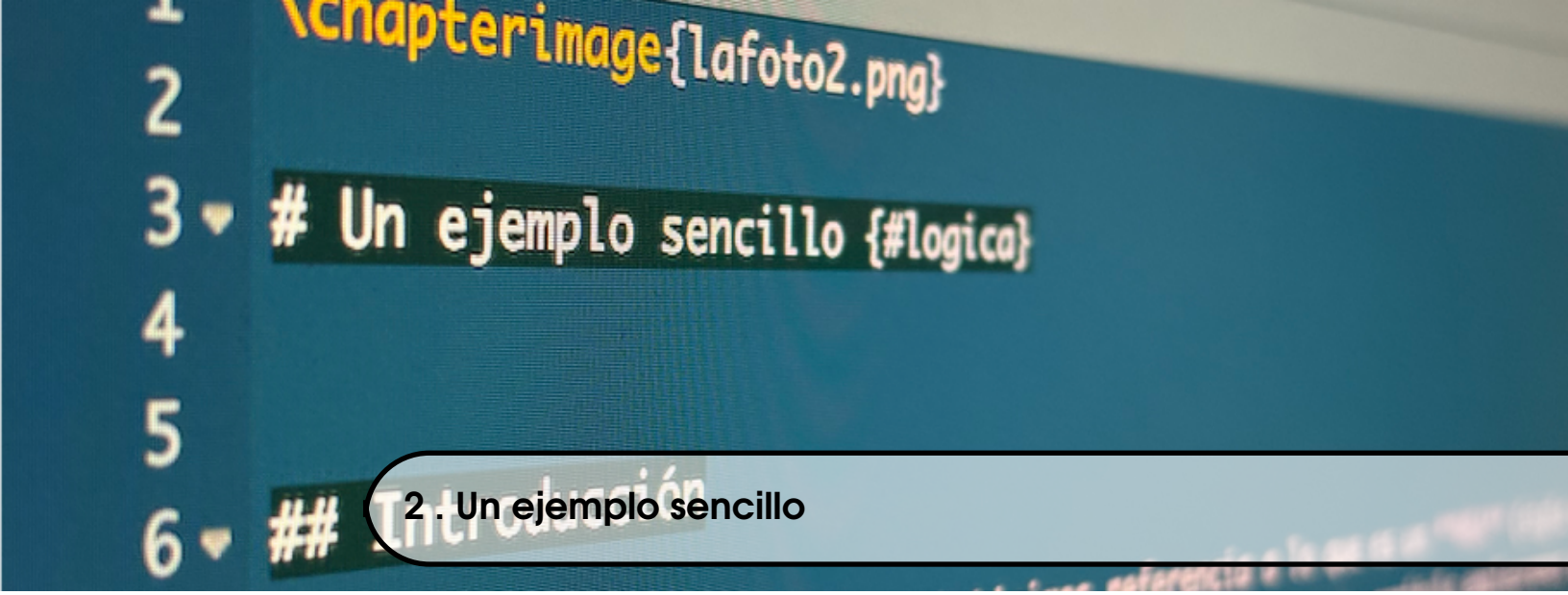
del consumidor dados los productos que ya se encuentran en un conjunto de productos. Esto lo discutiremos en detalle en el Capítulo 3.

La **analítica prescriptiva** busca responder la pregunta: ¿qué necesito hacer? El **MBA** nos permite hacer este tipo de analítica al sugerir claramente qué ítem será comprado dado que ya existe en la canasta otro bien. Esto lo discutiremos en detalle en el Capítulo 3.

1.4 Comentarios finales

En el Capítulo 2 discutiremos los principios del **MBA** por medio de un ejemplo sencillo. El ejemplo permite a lectores interesados en el rol de científico de datos como a lectores en el rol de *analytics translator* entender la lógica detrás de este análisis. En el Capítulo 3 se desarrolla un ejemplo con datos reales mostrando el paso a paso de un **MBA** empleando R.

Si bien el Capítulo 3 está dirigido al rol de científico de datos, es muy útil para el rol de *analytics translator* entender la "carpintería" detrás de un **MBA**. Como parte de un equipo, es importante que cada uno de los roles tenga un conocimiento general de los conceptos y de los procedimientos necesarios para el desarrollo de un **MBA**. Esto facilitará la colaboración y la comprensión mutua dentro del equipo. Finalmente, el Capítulo 4 discute las posibles formas de visualizar los resultados de un **MBA**, siendo este capítulo de igual importancia para ambos roles.



2 . Un ejemplo sencillo

2.1 Introducción

En el Capítulo 1 hicimos referencia a lo que es un **MBA** (sigla del término en inglés *Market Basket Analysis* o análisis de canasta). Sin embargo, aún debes tener muchas dudas acerca de qué es el **MBA**. En este capítulo emplearemos un ejemplo sencillo para aclarar conceptos fundamentales y explicar la lógica de las reglas de asociación del *business analytics*. De esta manera, este capítulo será de utilidad tanto para quien tiene el rol de científico de datos como para el *analytics translator*.

Al entender la lógica del proceso, es posible ejecutarlo (científico de datos) y saber qué decisiones se pueden tomar en el negocio (*analytics translator*) de acuerdo con las asociaciones propuestas. Este capítulo brinda los conceptos y el vocabulario que deben tener en común ambos roles para permitir la comunicación entre ellos. Las preguntas de negocio y la estrategia detrás de ellas serán comunicadas por el *analytics translator* a los científicos de datos. A su vez, los insights generados por el **MBA** realizado por los científicos de datos, que serán comunicados al *analytics translator*.

Por simplicidad, supongamos que estamos analizando datos de una tienda que solo vende cuatro productos: pan, leche, cerveza y pañales. Y con los datos queremos responder dos preguntas de negocio:

- La leche, al ser un producto perecedero que requiere refrigeración, puede implicar altos costos de almacenamiento y exhibición. Por eso, un gerente de tienda puede estar tentado a no ofrecer este producto, pero esto podría implicar que se dejen de vender otros productos. En este orden de ideas. La primera pregunta será: **¿se puede retirar del portafolio de la tienda la leche sin afectar las compras de otros productos?**
- Por otro lado, se acerca la fecha de expiración del lote de pan que está en la góndola y, antes de tener que botar el producto, el gerente del punto de venta quisiera ofrecer el pan en un combo a sus clientes. De esta manera, la segunda pregunta de negocio es: **¿qué producto debería acompañar al pan en un**

“combo” promocional?

Para realizar nuestro **MBA** será necesario primero contar con datos de transacciones (Sección 2.2); también requeriremos métricas de asociación de productos y conjuntos de estos (Sección 2.3); y finalmente es necesario contar con algoritmos para seleccionar las reglas más interesantes (Sección 2.4). Veamos estos elementos en detalle.

2.2 Los datos

Supongamos, para facilitar nuestro aprendizaje, que tenemos datos para cinco transacciones de nuestra tienda que solo vende cuatro productos: pan, leche, cerveza y pañales. En el Cuadro 2.1 se ve el reporte de los productos comprados en las cinco transacciones que tenemos. Noten que cada transacción (fila del Cuadro 2.1) solo tiene información del producto que está presente en la canasta de compra; los datos no incluyen ni precios de cada uno de los productos, ni cantidades compradas de cada producto.

Tabla 2.1. Productos comprados en cada transacción

ID	Productos
1	pan, leche
2	pan, cerveza, pañales
3	pan
4	pan, leche, cerveza, pañales
5	cerveza, pañales

Fuente: datos ficticios.

En la primera compra (`ID` = 1) se incluyeron los productos pan, leche. Esta compra se denomina transacción. Estos productos comprados corresponden a una tirilla de compra[1]. Una **canasta** se define como el conjunto de artículos que se adquieren en una compra (transacción). En este contexto, a un artículo o producto se le denomina ítem[2]. Esta definición permite emplear en este contexto el término carrito de compra como sinónimo de canasta.

Nota que la primera canasta está conformada por los ítems pan, leche. La segunda canasta contiene los ítems pan, cerveza, pañales. En el Cuadro 2.1 tenemos cinco canastas diferentes, cada una identificada con un `ID` diferente. Es importante reconocer

[1] Imagínate el recibo que te dan en un supermercado con el listado de los productos que compraste. Tirilla que incluye precio y número de unidades de cada producto.

[2] En algunas oportunidades escucharás el término SKU en vez de ítem. El SKU (la sigla de *stock keeping unit*) es el término técnico que se emplea en la gestión de inventarios para un tipo distinto de artículo para la venta o el inventario; en otras palabras, la referencia de un producto. Por ejemplo, el paquete de papas fritas de una marca en particular de 50 gramos tendrá un SKU y el paquete de 100 gramos del mismo tipo de papas fritas de la misma marca tendrá otro SKU y será tratado como si fuera otro ítem. En este libro no entraremos en ese detalle, pero nota que todo lo que estudiaremos aquí dependerá de qué definas como un ítem y esto dependerá de la pregunta de negocio. Es decir, si definimos como ítem las papas fritas o a un nivel más específico de papas fritas, los diferentes SKU para las papas fritas dependerán del problema bajo estudio.

que en este tipo de análisis nos importa la presencia del ítem en la transacción y no la cantidad del ítem comprado. Cuando tenemos datos que reportan transacciones, como los reportados en el Cuadro 2.1, se dice que contamos con datos transaccionales. Así, para hacer un **MBA** necesitaremos contar con datos transaccionales.

Regresando al Cuadro 2.1, podemos observar que los pañales fueron comprados con cerveza en tres de las cinco transacciones. Lo que hace que los pañales y la cerveza sean un conjunto de ítems frecuente (en inglés, *frequent itemset*). Adicionalmente, el pan fue el ítem más frecuentemente comprado.

Para formalizar un poco, se emplea la teoría de los conjuntos (una rama de las matemáticas que trabaja con conjuntos). En este caso se emplean los corchetes (`{}`) para enumerar los elementos de un conjunto y se emplea la coma (`,`) para separar cada elemento del conjunto. Así, la primera canasta se expresará como:

$$\{pan, leche\}$$

y la quinta canasta se representa como:

$$\{cerveza, pañales\}$$

Recordemos que en nuestro ejemplo, la tienda vende pan, leche, cerveza y pañales. Para hacer las cosas más fáciles, llamemos X al conjunto de todos los ítems que se pueden comprar en la tienda. Es decir,

$$X = \{pan, leche, cerveza, pañales\}$$

De esta manera, todas las cinco transacciones observadas en el Cuadro 2.1 corresponden a subconjuntos de X. Por ejemplo, la canasta tres es un subconjunto de tamaño uno (contienen un ítem de los cuatro ítems posibles). Las canastas uno y cinco son de tamaño dos.

El MBA en tu vida

Las películas o series que ves en *Netflix* u otro servicio de *streaming* se pueden también considerar como una canasta. De hecho, estas plataformas de *streaming* usan el **MBA** para sugerirte qué película o serie ver a continuación.

Las canciones que escuchas y tus *playlists* en *Spotify* también son consideradas como una canasta. *Spotify* utiliza algoritmos del **MBA** para recomendarte nueva música basada en tus preferencias reveladas en las canciones escuchadas y listas de reproducción anteriores.

Además, las compras que realizas en línea en plataformas como *Amazon* o *Mercado Libre* también emplean **MBA** para sugerir productos relacionados o complementarios a los que ya tienes en tu "carrito" de compra o en tus compras anteriores.

Todas estas aplicaciones son conocidas en el *business analytics* como **modelos de recomendación**. Los modelos de recomendación se definen como sistemas que sugieren ítems relevantes para los usuarios en función de sus preferencias o de consumidores similares y comportamientos pasados. Los modelos de recomendación pueden ser implementados empleando diferentes algoritmos que corresponden a diferentes "filosofías".

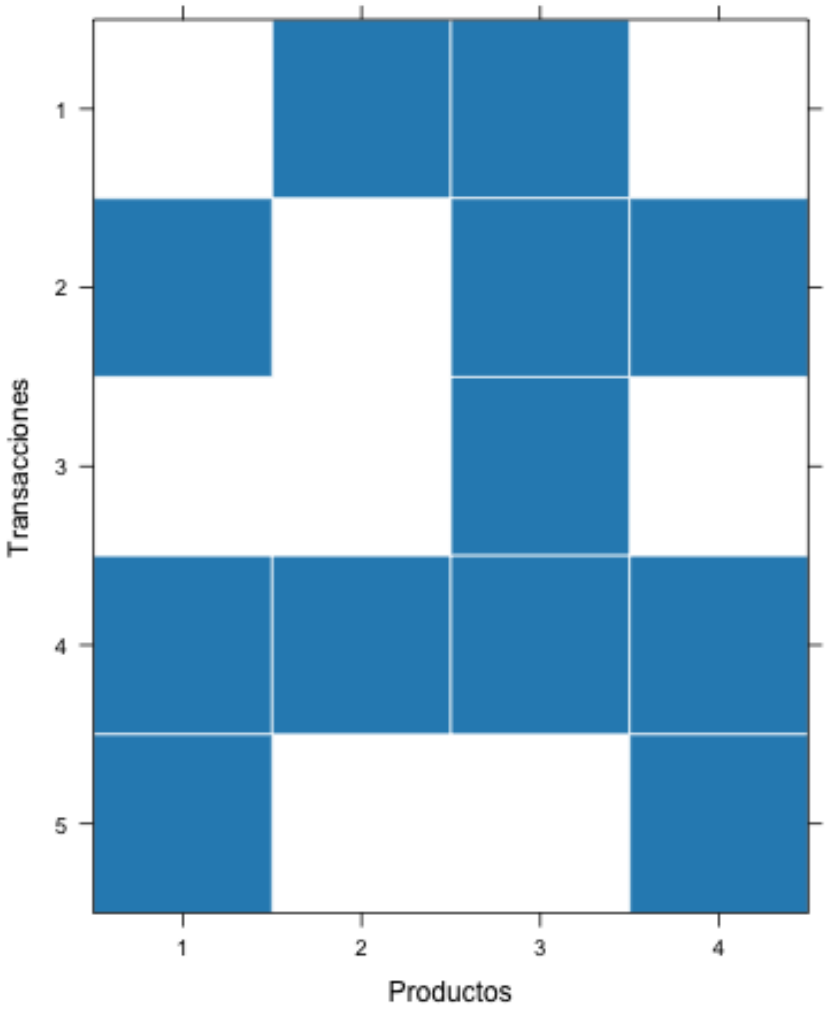
Estos algoritmos pueden clasificarse según los algoritmos que se emplean en aquellos basados en:

- Reglas de asociación (como las estudiadas en este libro): Estos algoritmos encuentran qué ítems son comprados con otros en la misma canasta y así sugieren aquellos ítems que aún no están presentes en la canasta del cliente. Estos algoritmos realizan la tarea de encontrar reglas de asociación.
- Filtros colaborativos basados en el usuario (*User-based collaborative filtering*): Estos algoritmos generan recomendaciones al encontrar usuarios similares a través de sus comportamientos pasados. Estos algoritmos realizan la tarea de formar clústeres.
- Filtros colaborativos basados en ítems (*Item-based collaborative filtering*): A diferencia de los algoritmos anteriores, estos emplean la similitud entre ítems basada en las valoraciones de los usuarios para encontrar ítems similares a los que le gustan al usuario. Estos algoritmos realizan la tarea de formar clústeres.
- Modelos de factores latentes (*Latent factor models*): Estos algoritmos encuentran las relaciones entre usuarios e ítems. Estos modelos asignan a cada usuario y a cada elemento un conjunto de factores latentes que capturan las preferencias y características relevantes. Estos algoritmos realizan la tarea de hacer regresiones.
- Ítems populares: Estos algoritmos no personalizados recomiendan a todos los usuarios los ítems más populares que aún no han valorado. Estos algoritmos realizan la tarea de resumir datos, al emplear estadísticas descriptivas para crear la recomendación.

Así, los algoritmos para encontrar reglas de asociación estudiados en este libro se pueden emplear para construir modelos de recomendación, pero no todos los modelos de recomendación emplean reglas de asociación.

Una manera de explorar los datos transaccionales es visualizándolos mediante una **matriz de ítems** (*item matrix*). La **matriz de ítems** permite tener una visión general de todas las transacciones y artículos al mismo tiempo. Es una visualización que en las columnas tiene los diferentes productos y en las filas cada una de las transacciones. Una celda oscura significa que el artículo pertenece a esa transacción, mientras que una celda blanca significa que el artículo no forma parte de la transacción. En la Figura 2.1 se presenta la *Matriz de ítems* para las transacciones de nuestro ejemplo registradas en el Cuadro 2.1.

Figura 2.1. Ítems por transacción



Fuente: elaboración propia. Nota: 1 = cerveza, 2 = leche, 3 = pan, 4 = pañales

Hay que tener cuidado al emplear esta visualización cuando se trabaja con millones de transacciones, pues es posible que tu computador se congele al tratar de hacer la visualización o, en el mejor de los escenarios, será muy difícil observar algo en ella.

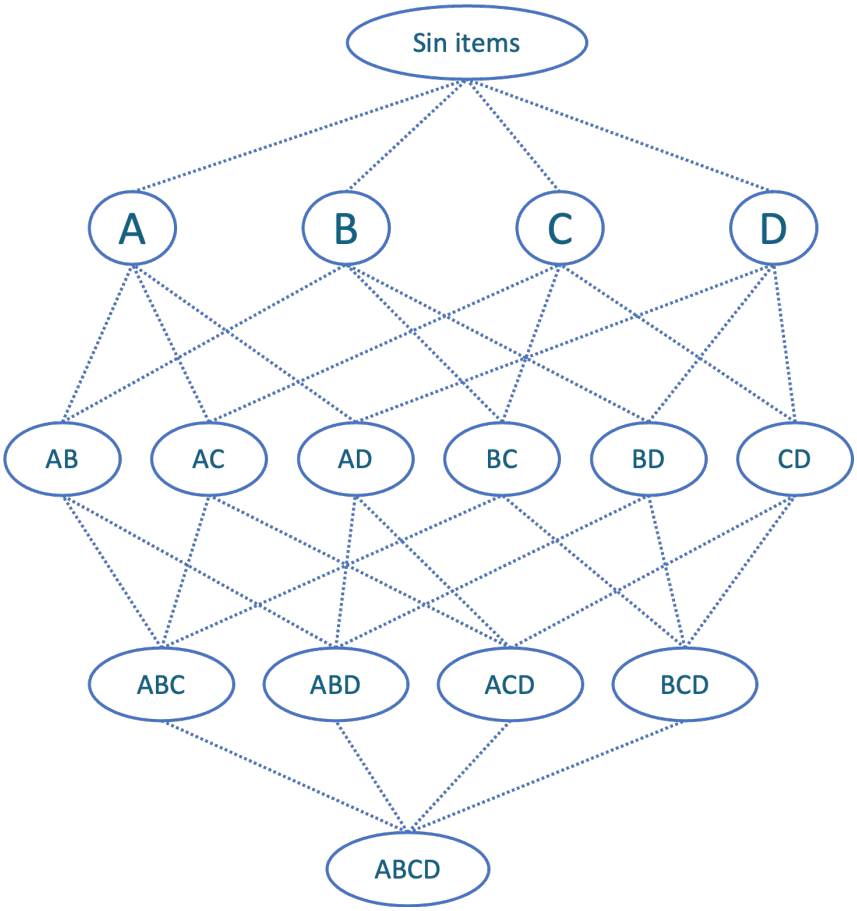
Este tipo de visualización proporciona una visión general de las transacciones y nos da indicios sobre la frecuencia con la que los artículos forman parte de las transacciones. De hecho, a partir de esta visualización se puede calcular una métrica conocida como *la densidad de la matriz de ítems*. Esta métrica es la relación entre las celdas sombreadas y el número total de celdas. En nuestro ejemplo tenemos 12 celdas de 20; es decir, una densidad del 60%. Una densidad del 100% implica que todas las canastas contienen todos los productos, mientras que una densidad del 0% implica que ninguna canasta contiene ningún producto.

Noten que, si bien la densidad de la matriz de ítems nos permite saber qué tan poblada es la matriz de ítems, esta métrica no nos permite determinar nada sobre la variedad de posibles cestas que se pueden construir. Solo nos muestra qué tanto se compran todos los ítems disponibles. Pero es importante tener cuidado cuando tengamos muchos ítems y transacciones; es posible que en esos casos la densidad no sea de interés para responder alguna pregunta de negocio.

De hecho, con los cuatro ítems que vende la tienda sería posible crear 16 canastas diferentes. En la Figura 2.2 se presentan todos los posibles subconjuntos que podemos crear de un conjunto X conformado por cuatro elementos (A, B, C y D). Es decir, $X = \{A, B, C, D\}$

En general, si tenemos n ítems, entonces podremos crear 2^n posibles canastas. Lo que hace que se aumente rápidamente el número de posibles canastas a medida que aumenta la cantidad de ítems disponibles. Esto hace el **MBA** mucho más retador cuando el número de productos disponibles en una tienda se hace grande. Para facilitar el estudio de las canastas y la selección de reglas, se han diseñado unas métricas que estudiaremos a continuación.

Figura 2.2. Todas las posibles canastas (subconjuntos) de un universo de 4 ítems.



Fuente: elaboración propia.

2.3 Métricas para itemsets y reglas de asociación

Una **regla de asociación** implica encontrar la relación entre un conjunto de productos que será probablemente comprado si ya está presente en la canasta un conjunto determinado de bienes. Por ejemplo, una regla de asociación podría ser: *si ya están en la canasta cerveza y leche, se comprará pan*. En general, se pueden construir reglas de asociación entre todos los (2^n) posibles subconjuntos. Pero esto no sería práctico dado que serán muchas reglas de asociación. Adicionalmente, algunas reglas de asociación serán irrelevantes al no ser observadas con frecuencia, es decir, al tener una poca probabilidad de ocurrencia. Por eso, en la práctica, queremos concentrar la atención solamente en reglas que sean potencialmente "relevantes". Para establecer qué regla puede ser relevante, se emplean típicamente tres métricas de asociación:

- soporte (*support* en inglés),
- confianza (*confidence* en inglés),
- cubrimiento (*coverage* en inglés) y
- *lift*.

A continuación veremos cada una de estas medidas.

2.3.1 Soporte (de un itemset)

El soporte representa la "popularidad" de un conjunto de artículos o **ítems**. En la jerga del **MBA**, un conjunto de artículos se conoce por el término en inglés *itemset*[3]. El soporte de un *itemset* Y cualquiera ($supp(Y)$) es la proporción de transacciones en las que aparece el conjunto de ítems$Y$ (como un subconjunto de cada transacción).

Formalmente, el soporte de un itemset Y se puede expresar como:

$$supp(Y) = \frac{freq(Y)}{N}$$

donde $freq(Y)$ es la frecuencia en que se observa el conjunto de productos Y y N es el número de canastas observadas en los datos.

Regresando a nuestro ejemplo, se reporta en el Cuadro 2.1, el *itemset* compuesto por solo el producto pan ($Y = \{pan\}$) está contenido (es un subconjunto) de 4 de las 5 canastas. Por eso, el *itemset* pan tiene un soporte de 80 % (4/5).

Tal vez sea más interesante conocer el soporte de *itemsets* que contengan más de un artículo. Por ejemplo, el *itemset* cerveza y pañales ($Y = \{cerveza, pañales\}$) tiene un soporte del 60 % (3 de 5 canastas). En el Cuadro 2.2 se muestran con negrita los *itemsets* que contienen cerveza y pañales en las tranasacciones observadas.

Nota que cuando hablamos del *itemset* cerveza y pañales ($Y = \{cerveza, pañales\}$), nos estamos refiriendo a un *itemset* que es resultado de la unión de los subconjuntos $A = \{cerveza\}$ y $B = \{pañales\}$. Es decir, $Y = A \cup B$.

Por otro lado, el *itemset* de pan y leche tiene un soporte de 40 % (2/5). De esta manera se puede calcular el soporte para todos los subconjuntos de canasta posibles.

[3]Nota que un *itemset* puede estar constituido por un solo ítem o por muchos de ellos.

Tabla 2.2. Canastas en las que aparece el itemset $Y = \{cerveza, pañales\}$ (en negrilla)

ID	Productos
1	pan, leche
2	**pan, cerveza, pañales**
3	pan
4	**pan, leche, cerveza, pañales**
5	**cerveza, pañales**

Fuente: elaboración propia.

Es decir, para los (2^n) posibles subconjuntos, que en este caso del ejemplo son 16 (Ver Figura 2.2).

En la práctica emplearemos el soporte para "filtrar" aquellas reglas que solo incluyan *itemsets* con un determinado soporte. Por ejemplo, reglas con *itemsets* que tengan como mínimo un soporte del 20%.

2.3.2 Confianza (de una regla)

La confianza corresponde al porcentaje de veces que se incluye en la canasta el *itemset* B dado que ya estaba en la canasta el *itemset* A. Es decir, es la probabilidad observada (frecuencia relativa) de la presencia del *itemset* B cuando ya estaba presente el *itemset* A. En otras palabras, estamos calculando una métrica para una regla de la forma **IFTTT**: "si pasa esto, entonces ocurre aquello". En este caso, si ya se tiene A en la canasta, entonces ocurre B. Esta **regla** la podemos expresar de la siguiente manera empleando la notación de la teoría de conjuntos:

$$A \rightarrow B$$

A es conocido como el antecedente de la regla y B como el consecuente. A también es conocida en la literatura del *MBA* como la condición que se encuentra a mano izquierda (**LHS** en inglés por *Left-Hand-Side*) y, de manera análoga, B es la condición de la mano derecha (**RHS** en inglés por *Right-Hand-Side*).

Regresando a esta métrica, **la confianza de la regla $A \rightarrow B$ ($conf(A \rightarrow B)$) se define como la probabilidad que se dé la regla $A \rightarrow B$ dado (condicional a) que A ya ocurrió**. Visto de otra manera, la confianza es el soporte de la unión de A y B dividido por el soporte de A. De manera formal,

$$conf(A \rightarrow B) = \frac{supp(A \cup B)}{supp(A)}$$

Si te sientes más familiarizado con la notación empleada en los cursos de estadística, la confianza se puede interpretar como una estimación de la probabilidad de que ocurra el *itemset* B dado que ya se observó el *itemset* A; formalmente, $P(B|A)$.

Por ejemplo, la confianza de la regla $cerveza \rightarrow pañales$ es de 1 o 100% ($supp(cerveza \cup pañales) = 3/5$ o 60% y $supp(cerveza) = 3/5$ o 60%). Esto quiere decir que, en nuestro

ejemplo, existe una probabilidad del 100% de observar pañales en la canasta, dado que ya se tiene cerveza en esta. Por otro lado, la confianza de la regla $pan \rightarrow leche$ es de 0.5 o 50% ($supp(pan \cup leche) = 2/5$ o 40% y $supp(pan) = 4/5$ o 80%). En otras palabras, según lo observado, cuando ya está el pan en la canasta, existe una probabilidad del 50% de que se incluya la leche.

También es posible calcular la confianza para *itemsets* que incluyan más de un producto. Por ejemplo, supongamos que queremos conocer la confianza de que se incluya pan en la canasta si ya se tiene cerveza y pañales en la canasta; es decir, la regla $\{cerveza, pañales\} \rightarrow pan$. En este caso, tenemos que el soporte del *itemset* $\{cerveza, pañales\}$ es del 60% (3 de 5 canastas) y el soporte del *itemset* $\{pan, cerveza, pañales\}$ es del 40% (2 de 5 canastas). Esto implica que la confianza de la regla $\{cerveza, pañales\} \rightarrow pan$ es 2/3 o 66.7%.

Nota que no necesariamente la confianza de la regla $A \rightarrow B$ será igual a la de la regla $B \rightarrow A$. Por ejemplo, intenta calcular la confianza de la regla $pan \rightarrow \{cerveza, pañales\}$.

Uno de los problemas que puede tener la confianza de la regla $A \rightarrow B$ es que esta depende fuertemente de la popularidad de A y no de B, y esto puede sesgar la importancia de una regla de asociación. Por ejemplo, la confianza de la regla $leche \rightarrow pan$ es de 1 o 100% por la baja popularidad de la leche. Recuerda que ya habíamos encontrado que la confianza de la regla $pan \rightarrow leche$ es de 50%.

Así como el soporte, la confianza se puede calcular para todas las posibles combinaciones disponibles y es empleada en la práctica para concentrar nuestra atención a reglas que tengan una confianza relativamente alta[4].

En otras palabras, el soporte y la confianza nos dan una medida de qué tan interesante puede ser un *itemset* y una regla, respectivamente. En un **MBA** las organizaciones[5] establecen unos umbrales mínimos de soporte y confianza de tal manera que se pueda comparar la fortaleza de unos conjuntos de artículos (*itemsets*) y reglas a la luz de las prácticas y riesgos que quiera asumir la organización.

2.3.3 Lift (de una regla)

Para resolver el problema que tiene la confianza y tener en cuenta la popularidad individual de ambos *itemsets*, se ha creado la métrica de elevación (más conocida por el término en inglés *lift*). El *lift* de una regla $A \rightarrow B$ es la confianza de esa regla dividida por el soporte del *itemset* A. Es decir,

$$lift(A \rightarrow B) = \frac{conf(A \rightarrow B)}{supp(B)}$$

[4]Si se tienen n *items*, entonces el total de *itemsets* que se pueden conformar es 2^n y el total de reglas de asociación son $3^n - 2^{r+1} + 1$. En el caso de nuestro ejemplo, $n = 4$. Así, el número de *itemsets* es de 16 y el número posible de reglas de asociación es de 50.

[5]Típicamente es una decisión que toma el *analytics translator*.

O dicho de otra manera, es el soporte del *itemset* $A \cup B$ relativo al soporte de A y B.

$$lift(A \rightarrow B) = \frac{supp(A \cup B)}{supp(A) \cdot supp(B)}$$

Esta última expresión implica que el *lift* de una regla es la probabilidad conjunta de que dos *itemsets* aparezcan juntos en una transacción, dividida por el producto de las probabilidades individuales de observar cada uno de los *itemsets* independientemente.

Intuitivamente, el *lift* representa la probabilidad de que el *itemset* B se compre cuando se compra el conjunto de artículos A, teniendo en cuenta la popularidad de B. Es la relación entre el soporte observado de la regla $A \rightarrow B$ y el soporte que se esperaría si la compra del *itemset* B fuera independiente del *itemset* B. Por eso el *lift* tiene una interpretación interesante:

- Si $lift(A \rightarrow B) = 1$, entonces no hay asociación entre los *itemsets* A y B (son independientes).
- Si $lift(A \rightarrow B) > 1$, entonces es probable que se compre el *itemset* B si se compra el A. En otras palabras, representa una asociación positiva entre los dos *itemsets*, lo que significa que los *itemsets* tienden a aparecer juntos con más frecuencia de lo esperado al azar.
- Si $lift(A \rightarrow B) < 1$, entonces es poco probable que se compre el *itemset* B si se compra el A. Es decir, existe una asociación negativa entre los dos *itemsets*, lo que significa que los *itemsets* tienden a aparecer juntos con menos frecuencia de lo esperado al azar.

En el caso de nuestro ejemplo, el *lift* de la regla $cerveza \rightarrow pañales$ es de $3/5 \approx 1{,}67$ ($conf(cerveza \rightarrow pañales) = 1$ y $supp(pañales) = 3/5$). Lo que implica que es probable que se compren pañales cuando se compra cerveza. Por otro lado, el *lift* de la regla $\{cerveza, pañales\} \rightarrow pan$ es $(2/3)/(4/5) = 5/6 \approx 0{,}83$. Es decir, es poco probable que se compre pan si en la canasta ya está incluido el *itemset* $\{cerveza, pañales\}$.

De esta manera, el *lift* es una métrica que nos ayuda a determinar si la combinación de un producto o productos con otro u otros mejora las posibilidades de realizar una venta. Además, también podemos descubrir si una regla de asociación no tiene ningún efecto o, peor aún, si es perjudicial.

2.3.4 Cobertura (de una regla)

Una medida que nos permite entender la frecuencia con la que se puede aplicar la regla es la cobertura (en inglés, *coverage* o *cover*). La cobertura también es conocida como el soporte de la mano izquierda (en inglés *Left Hand Side support* o *LHS-support*). Como lo indica este último nombre, la cobertura es el soporte del *itemset* que es antecedente de la regla o condición que se encuentra a la izquierda (**LHS**). Por ejemplo, en la regla $pan \rightarrow \{cerveza, pañales\}$ el antecedente es el *itemset* conformado por pan. Entonces, la cobertura de la regla $pan \rightarrow \{cerveza, pañales\}$ sería igual a $supp(pan) = 2/5$ o 40 %. En otras palabras, la regla $pan \rightarrow \{cerveza, pañales\}$ se podría aplicar con una probabilidad del 40 %.

2.4 Algoritmo para encontrar reglas

Hasta aquí hemos estudiado métricas que permiten caracterizar subconjuntos (*itemsets*) de productos como el soporte y reglas como la confianza, el *lift* y la cobertura. En la práctica, concentrar la atención en los 2^n posibles *itemsets* o $3^n - 2^{n+1} + 1$ posibles reglas de asociación será muy difícil. En el caso de nuestro sencillo ejemplo, una tienda con cuatro productos, estaríamos hablando de 16 *itemsets* y 50 reglas de asociación. Pero en el caso de un supermercado, fácilmente tendremos más de mil productos ($n = 1000$), lo que convierte el número de *itemsets* y reglas en una magnitud muy grande.

Por eso la pregunta natural es: **¿cómo descartar las reglas que no son pertinentes empleando estas métricas?** En otras palabras, necesitamos un algoritmo[6] que nos permita descartar rápidamente reglas que potencialmente no serán interesantes.

El algoritmo más sencillo (pero potente) y más usado para esta tarea se conoce como el algoritmo *Apriori*[7]. El algoritmo *Apriori* fue propuesto por Agrawal et al. (1994) como un algoritmo que parte de todos los *itemsets* que contienen un solo producto y va considerando de manera iterativa *itemsets* que contienen un elemento más. Este tipo de aproximación se conoce como una aproximación de abajo hacia arriba (*bottom-up* en inglés). La idea detrás de este algoritmo es muy sencilla. Si un *itemset* $\{A\}$ con un solo producto es poco frecuente, entonces todos los *itemsets* que lo contengan (súperconjuntos), como $\{A, B\}$, $\{A, C\}$ y $\{A, B, C\}$, serán también infrecuentes. De esta manera, podemos descartar reglas que contengan los *itemsets* de un producto que sean relativamente poco populares[8].

Así, el algoritmo *Apriori* implica los siguientes pasos:

1. Seleccionar todos los *itemsets* observados que tienen un solo producto.
2. Calcular el soporte de los *itemsets* seleccionados. Este paso se conoce como generación de candidatos.
3. Empleando el umbral aceptable para el soporte previamente establecido, descartar todos los *itemsets* seleccionados que no superen el umbral. Así mismo, descartar todos los *itemsets* más grandes que contengan los *itemsets* descartados.
4. Seleccionar los *itemsets* que tengan un producto más y realizar el paso 2 y 3 nuevamente.
5. Parar el algoritmo cuando no existan más *itemsets* para evaluar.

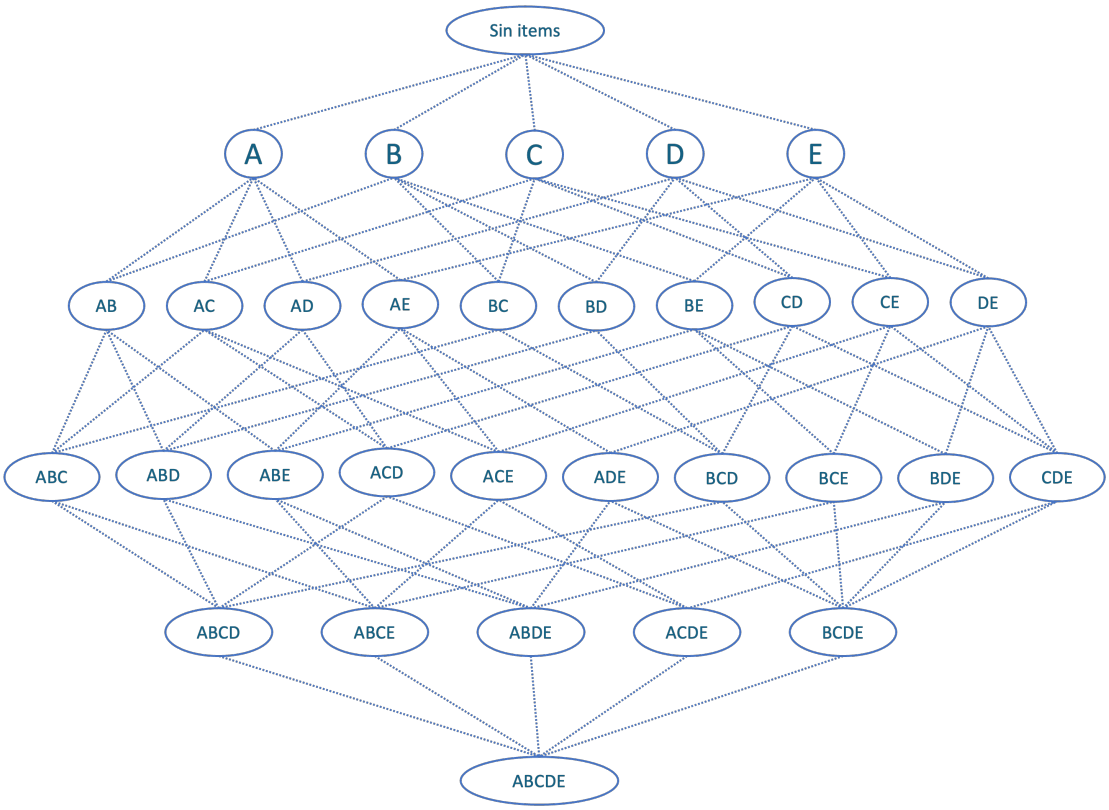
Veamos un ejemplo para entender la lógica de este algoritmo. Supongamos que un almacén tiene disponibles cinco ítems: (A, B, C, D y E). En la Figura 2.3 se presentan todos los posibles *itemsets*.

[6] Recuerda que un algoritmo es un conjunto ordenado de operaciones sistemáticas que permite hacer un cálculo y hallar la solución de un problema determinado.

[7] Existen otros algoritmos que se emplean para buscar reglas de asociación como AIS, SETM, FP-Growth, Eclat. Pero estos no los cubriremos en este libro.

[8] Nota que esta lógica también implica que, si un *itemset* con los elementos $\{A, B\}$ es frecuente, entonces ambos subconjuntos $\{A\}$ y $\{B\}$ son frecuentes.

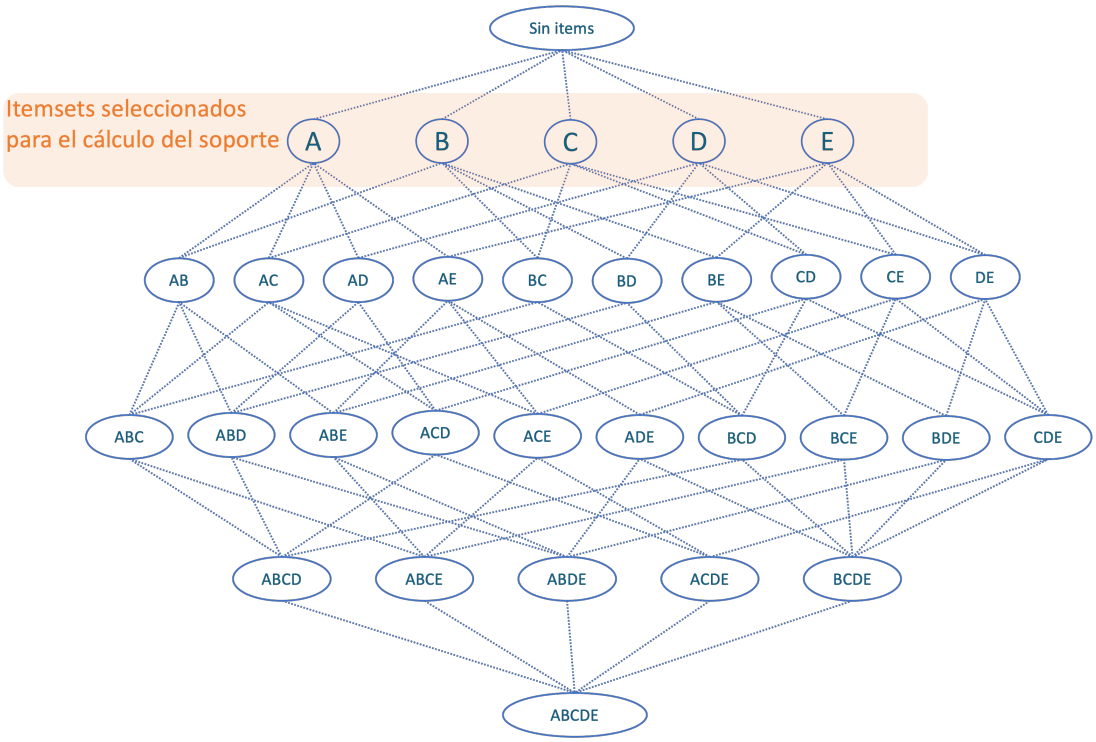
Figura 2.3. Todas las posibles canastas (subconjuntos) de un universo de 5 ítems.



Fuente: elaboración propia.

El primer paso del algoritmo implica empezar por los *itemsets* con un solo ítem como se muestra en la Figura 2.4.

Figura 2.4. Paso 1 del algoritmo A-Piori.

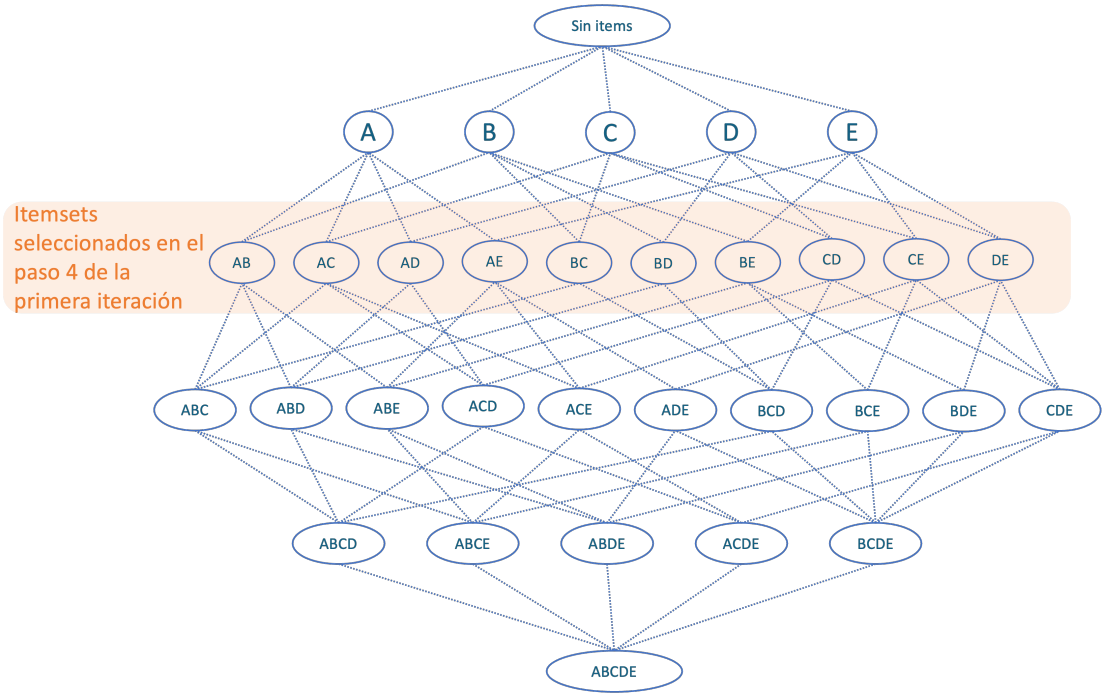


Fuente: elaboración propia.

El segundo paso implica calcular el soporte para cada uno de esos conjuntos. Ahora, supongamos que todos los *itemsets* tienen un soporte por encima del umbral establecido. Entonces, el tercer paso implica conservar todos los *itemsets*.

El cuarto paso implica seleccionar los *itemsets* que contengan un elemento más, como se muestra en la Figura 2.5. Y ahora repetiremos el paso 2 en la segunda iteración. Es decir, calcularemos el soporte de los nuevos *itemsets*.

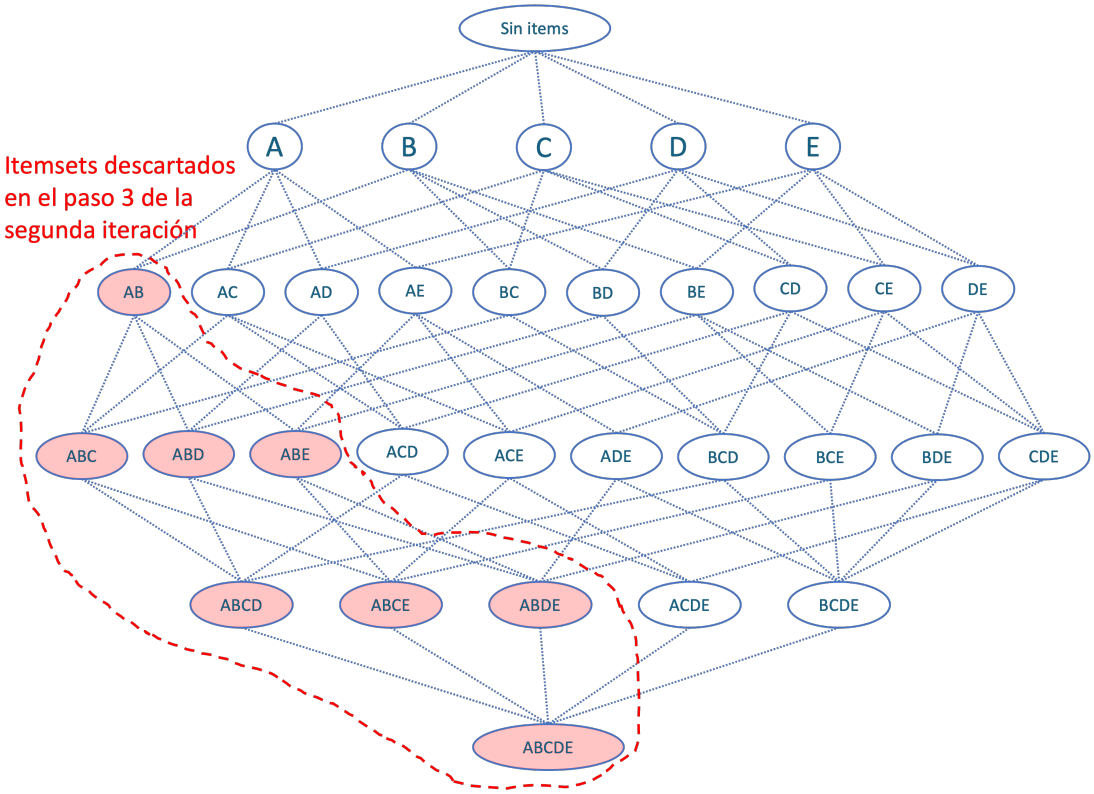
Figura 2.5. Paso 4 del algoritmo A-Piori en la primera iteración.



Fuente: elaboración propia.

Ahora, supongamos que el *itemset* $\{A, B\}$ tiene un soporte por debajo del umbral. Esto significa descartar todos los *itemsets* que contengan $\{A, B\}$. Tal como se presenta en la Figura 2.6.

Figura 2.6. Paso 3 del algoritmo A-Piori en la segunda iteración.



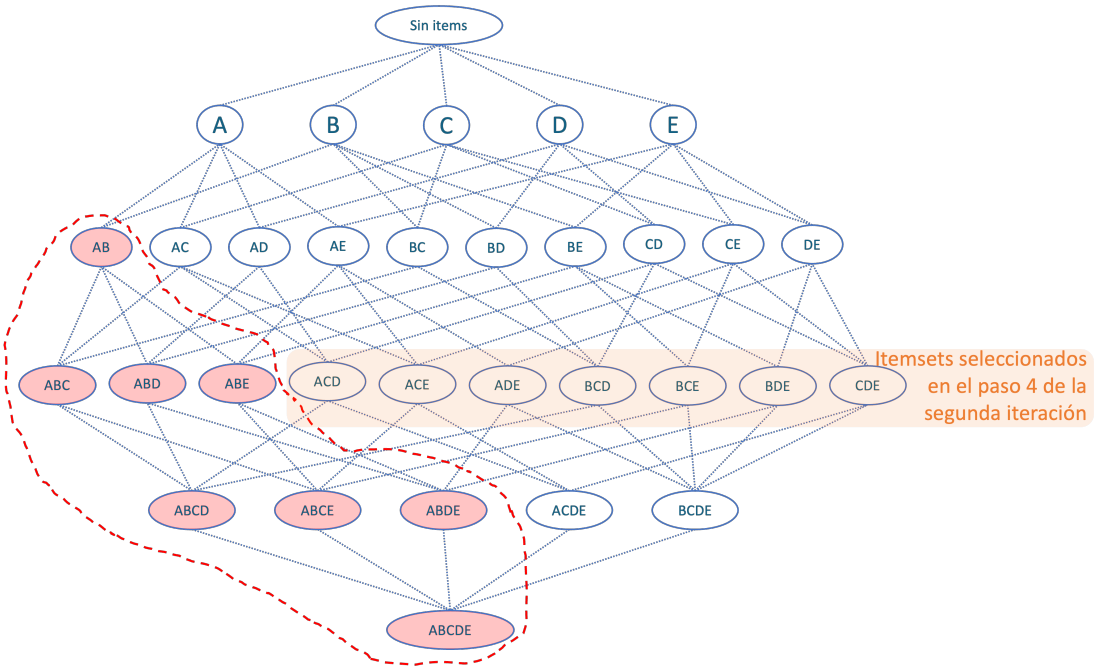
Fuente: elaboración propia.

Ahora, el paso 4 en la segunda iteración implica seleccionar los *itemsets* con un artículo más, tal como se muestra en la Figura 2.7.

Posteriormente, se repetiría el paso 2 al 4 hasta que se agoten los *itemsets*. Al final, con los *itemsets* que superan el umbral de soporte, se calcula la confianza, *lift* y cobertura para todas las reglas que se puedan establecer con los *itemsets* que quedan. Finalmente, el **MBA** continuará examinando las reglas disponibles para tomar decisiones de negocio.

Con este ejemplo, te puedes dar cuenta de cómo este algoritmo empieza rápidamente a reducir el número de *itemsets* y reglas que se analizarán, dejando solamente aquellas que cumplen el soporte (la popularidad) establecida como umbral. Nuestro ejemplo, por razones pedagógicas, emplea un número muy pequeño de ítems. Puedes imaginarte cómo este algoritmo facilita la vida cuando se cuenta con cientos o miles de productos, como ocurre en los almacenes de cadena.

Figura 2.7. Paso 4 del algoritmo A-Piori en la segunda iteración.



Fuente: elaboración propia.

Uno de los problemas del algoritmo *Apriori* es el costo computacional de la generación de conjuntos de elementos frecuentes. En ese proceso se necesita escanear la base de datos muchas veces, lo que conlleva un aumento del tiempo y reduce el rendimiento, en especial si la base de datos es relativamente grande.

Una de las decisiones más importantes para implementar este algoritmo es la definición del umbral para el soporte. Entre más bajo sea el umbral, menos *itemsets* serán descartados, pero entre más alto sea, podríamos estar descartando reglas relativamente razonables. Esa elección del umbral normalmente se basa en experiencia previa o ensayo y error. En general, es común empezar el análisis empleando un umbral del 10 % como valor inicial.

El algoritmo *Apriori* también puede ser empleado fijando como métrica para el umbral la confianza, el *lift* o una combinación de estos. De esta manera, al usar el algoritmo *Apriori* sobre un conjunto de transacciones, nuestro objetivo es identificar todas las reglas de asociación que tengan un valor igual o mayor al umbral de soporte, de confianza o *lift*.

Para nuestro ejemplo, si empleamos un umbral del 30 % para el soporte y uno de 50 % para la confianza, el algoritmo *Apriori* encuentra las cinco reglas que se presentan en el Cuadro 2.3. Las columnas "**LHS**" y "**RHS**" corresponden al antecedente y consecuente de cada una de las reglas, respectivamente, y las filas corresponden a cada una de las reglas identificadas por el algoritmo *Apriori* con los umbrales de soporte y confianza establecidos.

Tabla 2.3. Resultados de aplicar el algoritmo Apriori al ejemplo con soporte mayor a 30 % y confianza mayor a 50 %

LHS		RHS	Soporte	Confianza	Cobertura	Lift
cerveza	=>	pañales	0.6	1.00	0.6	1.67
pañales	=>	cerveza	0.6	1.00	0.6	1.67
leche	=>	pan	0.4	1.00	0.4	1.25
pan	=>	leche	0.4	0.50	0.8	1.25
cerveza	=>	pan	0.4	0.67	0.6	0.83
pan	=>	cerveza	0.4	0.50	0.8	0.83
pañales	=>	pan	0.4	0.67	0.6	0.83
pan	=>	pañales	0.4	0.50	0.8	0.83
cerveza, pañales	=>	pan	0.4	0.67	0.6	0.83
cerveza, pan	=>	pañales	0.4	1.00	0.4	1.67
pan, pañales	=>	cerveza	0.4	1.00	0.4	1.67

Fuente: elaboración propia.

El algoritmo *Apriori* encuentra la regla "cuando en la canasta está la cerveza, se compra pañales", que tiene una popularidad del 60 % (soporte) y una confianza del 100 % (ver primera fila del Cuadro 2.3). Es decir, el *itemset* cerveza y pañales ($\{cerveza, pañales\}$) fue observado en el 60 % de las transacciones, y en todos los

casos en los que la cerveza ya estaba en el "carrito", se incluyeron pañales en la transacción. Así mismo, el *lift* para esa regla es mayor que 1, lo cual implica que es muy probable que se compren pañales cuando en el "carrito" de compra tenemos cerveza. Nota que para nuestro ejemplo (no siempre será así) se obtienen los mismos resultados para la regla $pañales \rightarrow cerveza$. En este caso se detectaron once reglas.

2.5 Reglas que no agregan valor

No toda regla que encontramos con el algoritmo *Apriori* es útil o relevante para el negocio. Una buena regla no solo debe tener un equilibrio entre frecuencia (soporte), confianza, cubrimiento y *lift*, sino también que sean relativamente sencillas de accionar. Cuando se generan reglas de asociación, es común encontrar reglas que, aunque técnicamente distintas, expresan la misma relación de manera más compleja o menos eficiente. Estas son conocidas como **reglas redundantes**.

Una regla se considera redundante cuando su información ya está contenida en otra regla más simple o más general. A menudo, las reglas redundantes son más largas, incluyen más productos, pero no ofrecen una ganancia real en términos de soporte o confianza.

Para entender mejor este concepto, consideremos las siguientes reglas:

- $\{cerveza\} \rightarrow \{pan\}$ (quinta fila del Cuadro 2.3) y
- $\{cerveza, pañales\} \rightarrow \{pan\}$ (novena fila del Cuadro 2.3).

La segunda regla incluye un producto adicional en el antecedente (pañales), pero no mejora la confianza o el *lift* en relación con la primera. Si los pañales no contribuyen sustancialmente a hacer más fuerte la predicción de compra del pan, la segunda regla puede considerarse redundante.
Desde un punto de vista de comunicación y acción, es preferible conservar reglas más simples y directas, ya que son más fáciles de entender, comunicar y activar comercialmente.

Nota que también son redundantes las reglas:

- $\{cerveza, pan\} \rightarrow \{pañales\}$ (décima fila del Cuadro 2.3) y
- $\{pan, pañales\} \rightarrow \{cerveza\}$ (última fila del Cuadro 2.3).

En el Cuadro 2.4 se presentan las reglas de asociación después de descartar las reglas redundantes.

En términos formales, una regla de asociación $A' \rightarrow B$ se considera redundante si existe al menos otra regla $A \rightarrow B$ donde A es un subconjunto de A' ($A \subset A'$) y ambas tienen los mismos valores de soporte y confianza[9]. En otras palabras, la información que aporta $\{A'\} \rightarrow \{B\}$ está completamente contenida en una regla más general. La regla $\{A'\} \rightarrow \{B\}$ no mejora la capacidad predictiva. Dicho de forma intuitiva, una

[9]En la literatura se denomina súper regla a toda regla de asociación $A' \rightarrow B'$ tal que existe otra regla $A \rightarrow B$ con $A \subset A'$ y/o $B \subset B'$; es decir, la súper regla amplía el antecedente (*super antecedent rule*) o el consecuente (*super consequent rule*) de una regla más general manteniendo la misma estructura de predicción.

Tabla 2.4. Resultados de aplicar el algoritmo Apriori al ejemplo con soporte mayor a 30 % y confianza mayor a 50 % y eliminar reglas redundantes

LHS		RHS	Soporte	Confianza	Cobertura	Lift
cerveza	=>	pañales	0.6	1.00	0.6	1.67
pañales	=>	cerveza	0.6	1.00	0.6	1.67
leche	=>	pan	0.4	1.00	0.4	1.25
pan	=>	leche	0.4	0.50	0.8	1.25
cerveza	=>	pan	0.4	0.67	0.6	0.83
pan	=>	cerveza	0.4	0.50	0.8	0.83
pañales	=>	pan	0.4	0.67	0.6	0.83
pan	=>	pañales	0.4	0.50	0.8	0.83

Fuente: elaboración propia.

regla redundante es como repetir lo mismo con palabras de más. Si ya sabemos que "comprar cerveza implica comprar pan", añadir "y pañales" al antecedente sin aumentar la probabilidad de observar pan no agrega valor para las organizaciones. Solo agranda la lista de reglas y dificulta concentrar la atención en aquellas que permiten tomar decisiones. Es apenas natural que todas las reglas que no sean considerada redundantes, serán denominadas **reglas no redundantes**[10].

La identificación y eliminación de normas redundantes es una labor técnica responsabilidad del científico de datos. Sin embargo, el *analytics translator* también tiene un papel importante en este proceso de eliminación de reglas redundantes; este papel debe cooperar en la interpretación de las reglas filtradas, garantizando que el conjunto final de reglas mantenga su importancia para el negocio y no se pierdan descubrimientos potencialmente valiosos debido a una eliminación automática excesivamente rigurosa.

2.6 Respondiendo las preguntas de negocio

Regresemos a las preguntas de negocio originales para darles respuesta con los *insights* del **MBA**. La primera pregunta de negocio era: **¿se puede retirar del portafolio la leche sin afectar las compras de otros productos?** Esta pregunta surgió porque el gerente de la tienda estaba pensando en no ofrecer la leche para disminuir costos de almacenamiento y exhibición.

De las reglas identificadas, podemos centrar nuestra atención en aquellas cuyo antecedente (**LHS**) contenga la leche. En este caso tenemos la regla $\{leche\} \rightarrow \{pan\}$ (Ver la tercera fila del Cuadro 2.3), que tiene una alta probabilidad de que la leche

[10]Esto suena raro, pero formalmente también es necesario definir qué es una regla no redundante. Formalmente una **regla no redundante** se define como aquella para la cual las otras reglas se consideran súper reglas y al mismo tiempo las otras reglas tiene una confianza mas baja. En otras palabras, una regla de asociación $A \rightarrow B$ es no redundante cuando no existe otra regla $A' \rightarrow B$ (o $A \rightarrow B'$) tal que $A' \subset A$ (o $B' \subset B$) y cuya confianza y soporte sean mayores o iguales a los de la primera.

impulse el pan (*lift* superior a uno) y, siempre que se incluye leche en la transacción, se incluye pan (confianza del 100 %). Así, dejar de vender leche implicará que se dejará de vender pan; este es un riesgo que debería medir el gerente antes de sacar de su portafolio la leche. Se requerirá de un análisis más minucioso que cuantifique los ahorros en costos de sacar la leche y las pérdidas en ingreso por dejar de vender pan y leche.

La segunda pregunta era: **¿qué producto debería acompañar al pan en un "combo" promocional?** Esta pregunta estaba motivada por la necesidad de vender pan antes de su pronta fecha de expiración. Para responder esta pregunta, centremos la atención en reglas que en el consecuente (**RHS**) tengan pan:

1. $\{leche\} \rightarrow \{pan\}$ (tercera fila del Cuadro 2.4),
2. $\{cerveza\} \rightarrow \{pan\}$ (quinta fila del Cuadro 2.4) y
3. $\{pañales\} \rightarrow \{pan\}$ (séptima fila del Cuadro 2.4).

Las dos últimas reglas tienen un *lift* inferior a uno, lo cual implica que es poco probable que se compre el pan. Así, podemos descartar esas dos reglas. La regla $\{leche\} \rightarrow \{pan\}$ tiene un *lift* mayor a uno (1.25) y una confianza del 100 %. Así, tiene sentido hacer un combo para evacuar rápidamente el pan que incluya leche y pan.

De las reglas encontradas también podemos generar *insights* valiosos para tomar otras decisiones del negocio (¿se te ocurre alguna?). Los resultados del **MBA** combinados con ingenio pueden traer muchos beneficios a los establecimientos minoristas.

2.7 Implicaciones prácticas

El resultado de este análisis muestra que es esencial la interacción entre el *analytics translator* y el científico de datos. El tabajo conjunto de estas partes es evidente al a) determinar cuál es la condición, en este caso el producto, de la mano derecha (RHS) y de la mano izquierda (LHS); b) analizar los resultados de soporte, confianza y *lift* para la toma de decisiones. Las decisiones con base en estos resultados tendrán consecuencias sobre la rentabilidad del negocio.

La respuesta a las preguntas de si es posible retirar la leche del portafolio y qué producto sería un buen combo para el pan, responden a la idea de entender hábitos del consumidor y cómo estos se traducen en su comportamiento de compra. Al entender estos patrones de comportamiento es posible hacer una oferta de productos pertinente para el consumidor y generar mayor rentabilidad para el fabricante del producto y el canal minorista. La leche no es un producto estrella porque solo es el antecedente en uno de los tipos de canasta, pero tiene una coincidencia fuerte con la compra de pan, el cuál sí es un producto determinante en la compra de todos los productos en el punto de venta. Es decir, en este ejemplo el pan es un producto ancla y la leche es un producto de nicho.

El producto ancla [11] es un elemento esencial en la tienda porque es el que genera

[11] Un producto ancla es aquel que facilita la visita de consumidores al punto de venta porque es algo básí-

recordación y es atractivo para una gran masa de consumidores (Choi, 2018) (*producto ancla*). Mientras tanto el producto de nicho solo es atractivo para un exclusivo grupo de consumidores. Es decir, es un grupo pequeño pero con hábitos o características individuales que facilitan que ocurra la compra del producto A junto al producto B. Dado el tamaño pequeño y las características específicas de un segmento nicho, la tienda pueda aprovechar esta oportunidad para hacer actividades promocionales más efectivas al estar dirigidas específicamente a este grupo (ver Capítulo 1 *marketing de nicho*). En cuanto al producto ancla, algunas tiendas minoristas deciden ofrecer el producto ancla bajo su propia marca que el consumidor asocia a un precio bajo, aumentando la demanda por el producto (Al-Monawer et al., 2021)[12] (*marca propia*).

Las preguntas de este ejemplo con relación al pan y la leche esperan ser simples para mostrar cómo es posible utilizar el **MBA** en la toma de decisiones importantes para las marcas y los canales minoristas. No obstante, las aplicaciones del **MBA** no se restringen a este ejemplo o al contexto del mercadeo. El **MBA** se puede utilizar en diferentes áreas. En general, el **MBA** puede identificar patrones de comportamiento en grandes conjuntos de datos en los que importa la presencia o no de una característica, comportamiento o ítem, lo cual tiene aplicación en diferentes áreas.

Por ejemplo, en el campo de la salud, puede emplearse para detectar qué morbilidades están asociadas con otras (se presentan al mismo tiempo). Esto puede ayudar a los médicos a identificar pacientes que podrían estar en mayor riesgo de desarrollar ciertas enfermedades y tomar medidas preventivas.

En el campo de la seguridad, el **MBA** se utiliza para detectar patrones de comportamiento sospechosos en transacciones financieras. En este contexto, una transacción puede ser las actividades que se realizan en un día; por ejemplo: retiro en oficina, retiro en cajero, consignación y traslados. El **MBA** puede encontrar los *itemsets* y reglas comunes de tal manera que, si se observa un **itemset** atípico o una regla con poca confianza, puede marcarse como un comportamiento sospechoso.

En el ámbito de la logística y el transporte, el **MBA** se utiliza para optimizar las rutas de entrega y reducir los costos de transporte. Por ejemplo, si se detecta que ciertos productos tienden a ser comprados juntos con frecuencia, se pueden agrupar en la misma ubicación de almacenamiento para facilitar la preparación de pedidos.

2.8 Comentarios finales

En este capítulo discutimos los datos transaccionales que son empleados en el **MBA** y las métricas más comunes para caracterizar *itemsets* (soporte) y reglas de asociación

co en las necesidades de los consumidores en general y significa una compra de poco riesgo o desembolso bajo. Un producto ancla puede no ser el producto más vendido, pero sí es el que desencadena la visita o la compra de muchos otros.

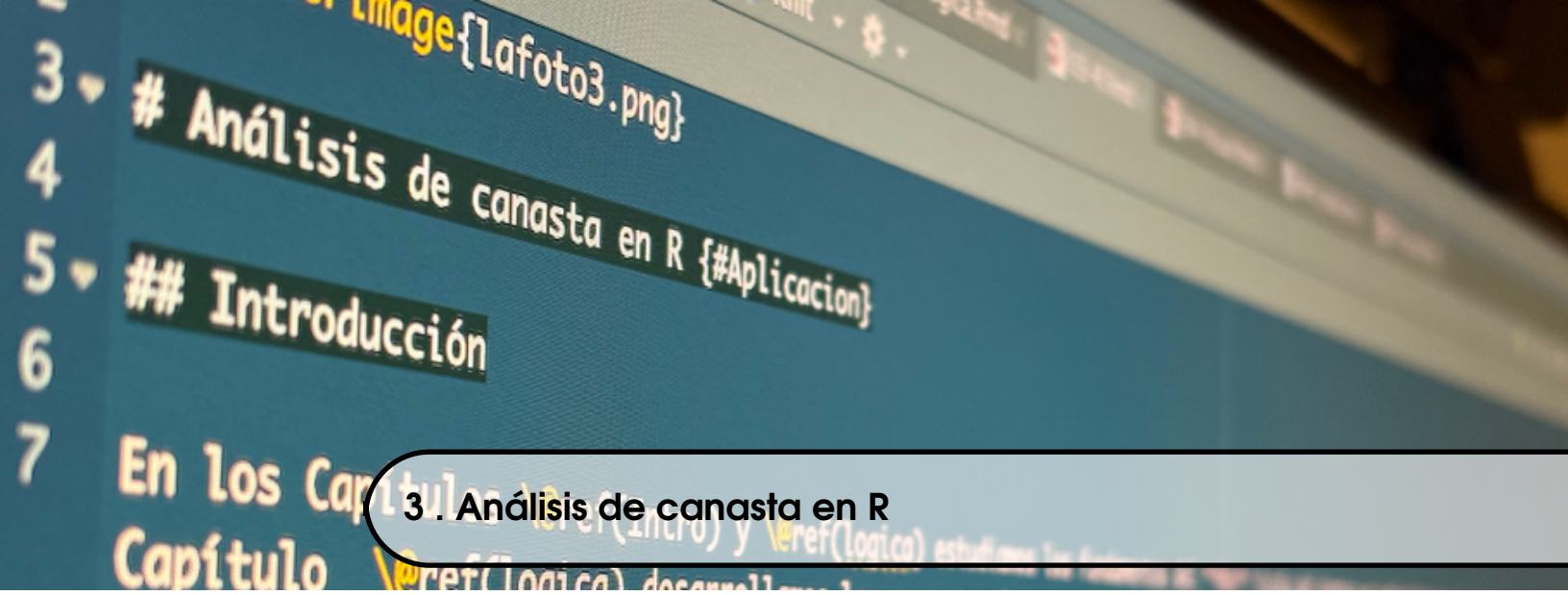
[12] Una marca propia es aquella que se distribuye bajo la etiqueta del canal minorista y no de un fabricante. Al funcionar como un producto genérico, en una determinada categoría, se ofrece a un menor precio pero con el respaldo de calidad dado por el distribuidor minorista

(confianza, *lift* y cobertura). Así mismo, estudiamos el algoritmo *Apriori*, que permite concentrar la atención sobre reglas "interesantes".

Al conocer la lógica del **MBA**, es claro que un uso efectivo requiere del trabajo conjunto de los roles del científico de datos y del *analytics translator*. El primero transforma los datos transaccionales y crea reglas de asociación empleando algoritmos; esto con un criterio coherente con las preguntas estratégicas del negocio. El segundo indica los lineamientos estratégicos desde el mercadeo y entiende los resultados para motivar decisiones prácticas que usen esas reglas de asociación.

Con un juego de datos transaccionales sencillo y empleando las herramientas del **MBA**, mostramos cómo encontrar reglas que podían responder las preguntas de negocio planteadas y permitir hacer analítica prescriptiva. Es decir, sugerir cuál es la mejor acción a tomar. En el área del *retail* (venta minorista), los resultados del **MBA** combinados con creatividad pueden traer muchos beneficios. Finalmente discutimos algunas implicaciones prácticas del ejercicio, destacando el potencial que tiene el **MBA** como herramienta de análisis en otros contextos.

Las aplicaciones del **MBA** son muchas, ¡la imaginación es el límite!



3 . Análisis de canasta en R

3.1 Introducción

En los Capítulos 1 y 2 estudiamos los fundamentos del **MBA** (sigla del término en inglés *Market Basket Analysis* o análisis de canasta). En el Capítulo 2 desarrollamos los conceptos fundamentales y empleamos un ejemplo sencillo para aplicar dichos conceptos. En este Capítulo nos concentraremos en aplicar los conceptos aprendidos en R (R Core Team, 2023). En la práctica, el **MBA** se realiza con bases de datos que registran millones de transacciones de establecimientos que cuentan con miles de productos (*ítems*).

El procesamiento de estas grandes bases de datos requiere de herramientas especializadas para su análisis. R es un lenguaje de programación ampliamente utilizado en el análisis de datos y ofrece diversos paquetes y funciones para realizar el **MBA**. En este Capítulo, exploraremos cómo emplear R y el paquete *arules* (Hahsler et al., 2011) para realizar un **MBA**. Con este objetivo, este Capítulo explica cómo preparar los datos para ser procesados (Sección 3.3), cómo emplear el algoritmo *Apriori* para la selección de reglas interesantes (Sección 3.4) y cómo trabajar con las reglas encontradas (Sección 3.5).

Este Capítulo explica al científico de datos los pasos para generar las reglas de asociación. Además es útil para el *analytics translator* para entender la lógica que sigue el científico de datos y poder guiar al científico de datos al crear los parámetros de asociación que se esperan visualizar. El *analytics translator* puede omitir de este Capítulo el detalle del código en R. Es decir, es importante que en el proceso exista un trabajo conjunto entre el científico de datos y el *analytics translator* para asegurarse de que los resultados sean coherentes con las preguntas del negocio (ver, por ejemplo, la sección 1.1) y las decisiones que se esperan tomar.

En el Capítulo 4 exploraremos cómo visualizar los resultados tanto de las métricas como de las reglas en sí, resultado del **MBA**. En el Capítulo 5 presentaremos un ejercicio de **MBA** completo en el que nos concentraremos más en las implicaciones del

negocio y una aproximación diferente en la exploración de los datos que tiene sentido en determinados negocios. Pero antes de entrar en esos detalles, aprenderemos, a continuación, las funciones y paquetes de R que permiten hacer un **MBA**.

3.2 Los datos

Para desarrollar nuestro ejemplo con datos reales, emplearemos un conjunto de datos que contiene transacciones de una empresa de comercio electrónico con sede en el Reino Unido provistas por ret (2015). Dicha empresa se caracteriza por la venta al por menor en línea de regalos para toda ocasión. La base de datos que emplearemos es un subconjunto de todos los datos transaccionales publicados por ret (2015); este subconjunto corresponde únicamente al primer trimestre de 2011 y se descartaron dos variables (la fecha de la transacción y el país de origen de la transacción). Los datos están disponibles en la página web del libro en el archivo `Online_Retail_data.csv`.

Cada fila del archivo corresponde a un ítem comprado y las variables incluidas en estos datos son las siguientes:

- `InvoiceNo`: Número de la factura que corresponde a un número entero de 6 dígitos. Esta variable permite identificar cuáles ítems (filas) corresponden a la misma transacción (canasta de compra o carrito de compra).
- `SKU`: código (número entero de 5 dígitos) del ítem (la referencia del ítem).
- `Description`: Descripción del ítem.
- `Quantity`: La cantidad comprada del ítem en la transacción.
- `UnitPrice`: Precio por unidad del ítem en libras esterlinas.
- `CustomerID`: Código (número entero de 5 dígitos) de identificación del cliente.

La base contiene 2981 ítems diferentes. Es decir, la tienda en línea, durante ese período, vendió 2981 diferentes productos (SKU diferentes) que implican un número muy grande de posibles *itemsets*[1] y de reglas[2]. En total se cuentan con 3777 transacciones de 1777 clientes diferentes. Nota que la base tiene 97449 líneas de datos; cada línea es un ítem vendido cuyos atributos (variables) son `SKU`, `Description`, `Quantity`, `UnitPrice` y `CustomerID`.

Para empezar, descarga el archivo `Online_Retail_data.csv` de la página web del libro, carga los datos y constata que los datos han sido leídos correctamente.

```
# Cargar los datos
Datos_original <- read.csv("Online_Retail_data.csv", sep = ",")

# Cargar paquete
library(dplyr)

# una miras rápida a los datos cargados
glimpse(Datos_original)
```

[1] ¡Intenta calcular 2^{2981} en R! Ese es el número máximo de itemsets posibles.
[2] ¡Intenta calcular $3^{2981} - 2^{2981+1} + 1$ en R! Ese es el número máximo de reglas posibles.

```
## Rows: 97,449
## Columns: 6
## $ InvoiceNo   <int> 539993, 539993, 539993, 539993, 539993, 539993, 539993, 53~
## $ SKU         <chr> "22386", "21499", "21498", "22379", "20718", "85099B", "20~
## $ Description <chr> "JUMBO BAG PINK POLKADOT", "BLUE POLKADOT WRAP", "RED RETR~
## $ Quantity    <int> 10, 25, 25, 5, 10, 10, 6, 12, 6, 8, 6, 6, 6, 12, 12, 8, 4,~
## $ UnitPrice   <dbl> 1.95, 0.42, 0.42, 2.10, 1.25, 1.95, 3.25, 1.45, 2.95, 1.95~
## $ CustomerID  <int> 13313, 13313, 13313, 13313, 13313, 13313, 13313, 13313, 13~
```

Antes de continuar, carga el paquete *arules* (Hahsler et al., 2011); si aún no lo has instalado, instálalo.

```
# Instalar paquete si aún no se tiene install.packages('arules')

# Cargar paquete
library(arules)
```

3.3 Preparación de los datos y análisis preliminar

Para emplear los datos con las funciones del paquete *arules*, es necesario convertirlos a la clase **transactions**; esta clase es exclusiva de este paquete. Para convertir un objeto de clase **data.frame** a **transactions**, tendremos que agrupar los ítems por el ID de la transacción. Es decir, crear las canastas o "carritos" de compra. Para realizar esta operación de manera rápida debemos seguir dos pasos. Primero, convertir los datos en clase **list** y el segundo paso es convertir la lista a clase **transactions**.

Para el primer paso, podemos emplear la función **split()** de la base de R. Esta función construirá un objeto de clase **list** [3] con diferentes compartimientos (*slots*), uno para cada transacción. La función **split()** tiene los siguientes argumentos:

split(x, f)

donde:

- **x**: Es vector o **data.frame** que contiene los valores a dividir en grupos.
- **f**: Es un vector de clase **factor** que definirá cómo agrupar los datos.

En nuestro caso queremos dividir los datos de tal manera que a cada transacción le queden sus correspondientes productos comprados. Es decir, la columna que contiene la descripción de los productos (variable `Description`) la queremos dividir[4] de tal manera que tengamos grupos por factura (`InvoiceNo`). Esto lo podemos hacer con el siguiente código:

```
# Separar de los datos
datos_lista = split(Datos_original$Description, Datos_original$InvoiceNo)
```

[3] Para una descripción de esta clase puedes consultar Alonso y Ocampo (2022).

[4] También podríamos emplear el SKU, pero de pronto los números de los SKU no nos dirán mucho. Por eso emplearemos mejor la descripción.

```
# Chequear la clase del objeto
class(datos_lista)
```

```
## [1] "list"
```

```
# Mirar el primer elemento de la lista
head(datos_lista, 1)
```

```
## $`539993`
##  [1] "JUMBO BAG PINK POLKADOT"            "BLUE POLKADOT WRAP"
##  [3] "RED RETROSPOT WRAP "                "RECYCLING BAG RETROSPOT "
##  [5] "RED RETROSPOT SHOPPER BAG"          "JUMBO BAG RED RETROSPOT"
##  [7] "RED RETROSPOT CHILDRENS UMBRELLA"   "JAM MAKING SET PRINTED"
##  [9] "RECIPE BOX RETROSPOT "              "CHILDRENS APRON APPLES DESIGN"
## [11] "PEG BAG APPLES DESIGN"              "COFFEE MUG APPLES DESIGN"
## [13] "COFFEE MUG PEARS  DESIGN"           "WHITE HANGING HEART T-LIGHT HOLDER"
## [15] "SET OF 6 T-LIGHTS EASTER CHICKS"    "CAST IRON HOOK GARDEN FORK"
## [17] "LOVE HEART NAPKIN BOX "
```

En el objeto `datos_lista` tenemos una lista en la que cada compartimiento (*slot*) corresponde a una transacción y contiene un "carrito" de compra. Por eso se tienen 3777 compartimientos.

Ahora podemos proceder al segundo paso: crear el objeto de clase **transactions**. Esto lo podemos hacer con la función **as()**. Esta función (del paquete *arules*) solo necesita como argumentos un objeto de clase **list** y especificar la clase a la que queremos convertir el objeto. En nuestro caso tendremos:

```
# Convertir a clase transactions
datos_tra <- as(datos_lista, "transactions")

# Chequear la clase del objeto
class(datos_tra)
```

```
## [1] "transactions"
## attr(,"package")
## [1] "arules"
```

Para inspeccionar los datos de clase **transactions** no podemos emplear la aproximación tradicional. Explora lo que ocurre si empleamos la función **head()** o simplemente tratamos de imprimir el objeto.

```
# Llamar al objeto
datos_tra
```

```
## transactions in sparse format with
##  3777 transactions (rows) and
##  2929 items (columns)
```

```
# Inspeccionar el objeto de la manera tradicional
head(datos_tra)
```

```
## transactions in sparse format with
##  6 transactions (rows) and
##  2929 items (columns)
```

```
# Imprimir en consola el objeto
print(datos_tra)
```

```
## transactions in sparse format with
##  3777 transactions (rows) and
##  2929 items (columns)
```

Para inspeccionar los datos, podemos emplear la función **summary()** del paquete *arules*. Recuerda que cuando tenemos funciones que tienen el mismo nombre en diferentes paquetes, típicamente estas están diseñadas para reaccionar diferente de acuerdo a la clase del objeto que se use en el argumento.

En nuestro caso, el código sería el siguiente:

```
# Resumir el objeto de clase transacctions
summary(datos_tra)
```

```
## transactions as itemMatrix in sparse format with
##  3777 rows (elements/itemsets/transactions) and
##  2929 columns (items) and a density of 0.00866279
##
## most frequent items:
## WHITE HANGING HEART T-LIGHT HOLDER  SET OF 3 CAKE TINS PANTRY DESIGN
##                                502                                481
##           REGENCY CAKESTAND 3 TIER            JUMBO BAG RED RETROSPOT
##                                457                                414
##  SET OF 6 SPICE TINS PANTRY DESIGN                            (Other)
##                                351                              93630
##
## element (itemset/transaction) length distribution:
## sizes
##   1   2   3   4   5   6   7   8   9  10  11  12  13  14  15  16  17  18  19  20
## 411 135 139 120 140  87 101 115 116  98 118  86  84  92  86 113  84  79 103  74
##  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40
##  76  73  61  56  33  42  47  46  53  44  35  28  29  36  26  30  17  21  31  19
##  41  42  43  44  45  46  47  48  49  50  51  52  53  54  55  56  57  58  59  60
##  29  24  18  22  15  17  13  23  20  15  13  16  16  12  11   8   9  11   5  12
##  61  62  63  64  65  66  67  68  69  70  71  72  73  74  75  76  77  78  79  80
##   7   3   8   9   8   5   7   6   9   7   8   4   6   5   5   4   2   7   3   1
##  81  82  83  84  85  86  87  88  89  90  92  93  95  96  97  98 101 104 106 107
##   2   4   1   1   3   2   1   3   6   3   1   3   2   2   1   4   3   3   3   2
```

```
## 108 109 110 111 113 114 116 117 118 119 120 122 124 126 127 129 130 131 135 138
##   1   2   1   3   1   1   1   1   1   1   1   1   1   1   2   1   2   1   1   2
## 139 140 141 143 144 147 149 151 153 154 155 156 157 158 159 162 165 166 168 169
##   1   1   1   1   3   2   3   2   2   1   1   1   1   1   1   1   2   2   2   2
## 171 172 174 175 177 178 180 182 185 186 189 192 193 198 199 204 206 208 214 215
##   1   1   1   1   3   2   2   2   1   1   1   1   1   1   1   2   1   2   1   1
## 216 219 222 223 224 225 228 230 235 250 271 280 285 288 298 332 333 339 345 348
##   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1
## 350 354 358 370 376 382 400 404 408 414 416 428 454 458 500
##   1   1   1   1   1   1   2   1   1   1   1   1   1   1   1
##
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    1.00    5.00   15.00   25.37   29.00  500.00
##
## includes extended item information - examples:
##                          labels
## 1
## 2  4 PURPLE FLOCK DINNER CANDLES
## 3     OVAL WALL MIRROR DIAMANTE
##
## includes extended transaction information - examples:
##   transactionID
## 1        539993
## 2        539997
## 3        539998
```

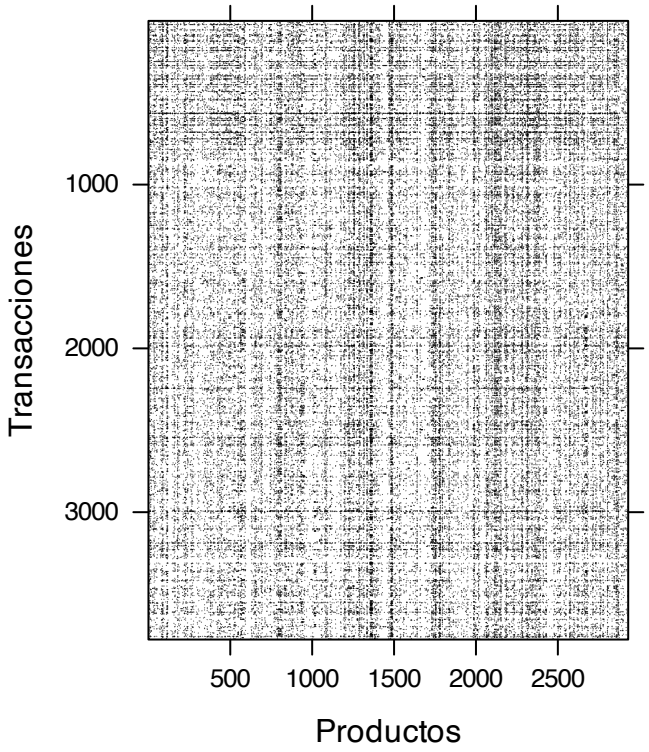
En este resumen, podemos encontrar información interesante. Por ejemplo:

- Contamos con 3777 transacciones (carritos de compra) y 2929 items.
- La densidad es de 0.866 %. Densidad muy baja si la comparamos con nuestro ejemplo del Capítulo 2. Pero este resultado es normal dada la gran cantidad de productos disponibles. En general, para este tipo de bases de datos de estos tamaños es común observar densidades de esta magnitud.
- También podemos ver que los ítems más frecuentes en las canastas son: WHITE HANGING HEART T-LIGHT HOLDER, SET OF 3 CAKE TINS PANTRY DESIGN , REGENCY CAKESTAND 3 TIER, JUMBO BAG RED RETROSPOT, SET OF 6 SPICE TINS PANTRY DESIGN, (Other).

El paquete *arule* permite construir rápidamente visualizaciones como la matriz de ítems (*item matrix*) que se discutió en la Sección 2.2. La función **image()** solo necesita como único argumento un objeto con las transacciones (de clase **transactions**) para crear la matriz de *ítems*. Esta función tiene más argumentos que permiten personalizar el gráfico, como **xlab** y **ylab** para cambiarle los nombres a los ejes. El siguiente código genera la Figura 3.1:

```
# Visualizar la matriz de items
image(datos_tra, xlab = "Productos", ylab = "Transacciones")
```

Figura 3.1. Items por transacción (matriz de items)



Fuente: elaboración propia.

La Figura 3.1 muestra lo que se intuía con el resultado de la densidad. Es una matriz poco poblada. Con este tipo de visualizaciones tenemos que tener mucho cuidado. Si mostramos demasiadas transacciones a la vez (por ejemplo, con millones de transacciones), esta visualización típicamente no informa mucho y puede tomar mucho tiempo y demandar muchos recursos de tu máquina e incluso bloquear tu computador. En este caso no parece agregar mucho valor esta visualización. ¡Siempre tenemos que intentarlo! No sabremos si una visualización funciona o no hasta no intentarlo y verla.

El paquete *arules* también incluye una función que permite visualizar los ítems (*itemsets* de un solo elemento) con mayor popularidad. La función **itemFrequencyPlot()** tiene como argumentos esenciales un objeto con las transacciones (de clase **transactions**) y el número de productos con la mayor frecuencia que se quieren visualizar (**topN**). Por ejemplo, hagamos un gráfico para el top 10 de ítems de mayor frecuencia. Esto se puede lograr con el siguiente código:

```
# Visualizar el top 10 de items más populares
itemFrequencyPlot(datos_tra, topN = 10)
```

El resultado de evaluar esta línea de código no se presenta para ahorrar espacio.

Existen otros argumentos de la función que nos facilitan continuar personalizando la visualización. Por ejemplo, el argumento **type** nos permite que el gráfico de barras se presente con la frecuencia relativa[5] (**type = "relative"**) o absoluta (**type = "absolute"**)[6]. También podemos hacer que las barras sean horizontales con el argumento **horiz = TRUE**. En la Figura 3.2 se presenta otra versión del gráfico que acabas de realizar. ¡Intenta replicar esta visualización!

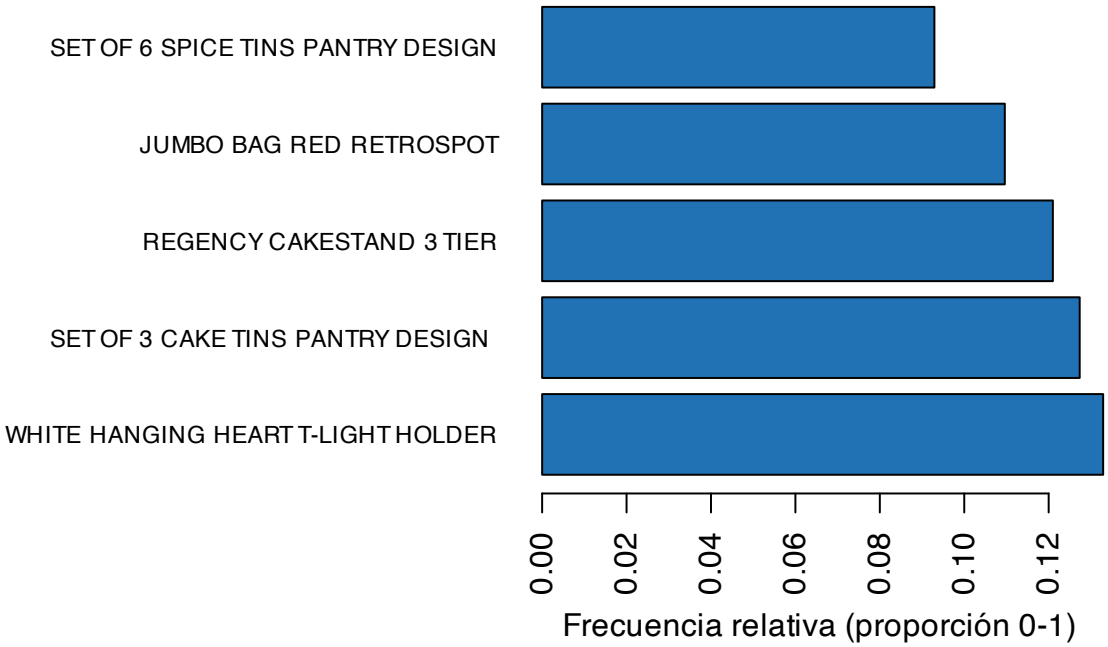
La Figura 3.2 muestra los mismos resultados que ya habíamos identificado en el resumen, pero de una manera más amigable. No obstante, las visualizaciones del paquete *arules* emplean la base de R. Ya sabemos que el paquete *ggplot2* (Wickham, 2016) genera visualizaciones más flexibles[7]. Podemos construir rápidamente una función que emplee los paquetes *ggplot2*, *tidyverse* (Wickham et al., 2019) y *arules* para generar una visualización similar.

A continuación se presenta una función que crea un objeto de clase *ggplot* al que se le pueden adicionar capas como queramos. Esta función es una versión levemente modificada de lukeA (2017).

```
itemFrequencyGGPlot <- function(x, topN, color) {
    require(tidyverse)
    require(ggplot2)
    require(arules)
    x %>%
        # Calcular las frecuencias relativas para cada ítem
    itemFrequency %>%
        # Organizar
```

[5] Es decir, como proporción.
[6] Es decir, como el número de veces observado.
[7] Si deseas recordar cómo funciona el paquete *ggplot2* puedes consultar Alonso y Largo (2023).

Figura 3.2. Los 5 productos con mayor frecuencia en las transacciones



Fuente: elaboración propia.

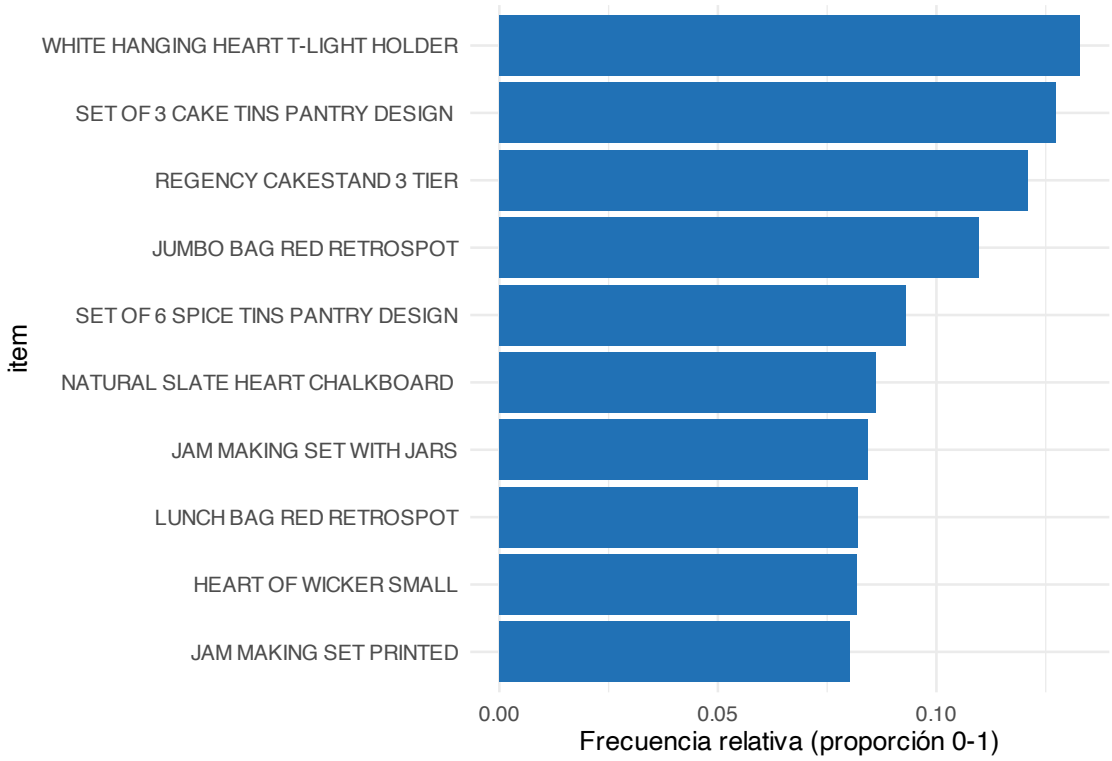
```
    sort %>%
        # Extraer los items deseados
    tail(topN) %>%
        # Crear data.frame
    as.data.frame %>%
        tibble::rownames_to_column() %>%
        # Generar la visualización
    ggplot(aes(reorder(rowname, .), .)) + geom_col(fill = color) + coord_flip()
}
```

En la Figura 3.3 se presenta una visualización creada con esta función. ¡Intenta replicarla!

Continuando con la exploración de los datos, en algunas ocasiones podemos querer inspeccionar algunas canastas. Esto lo podemos hacer con la función **inspect()** del paquete *arules*. Con esta es posible inspeccionar un grupo de transacciones o una en específico. Por ejemplo, inspeccionemos las dos primeras canastas con la siguiente línea de código:

```
# Inspeccionar las dos primeras transacciones
inspect(head(datos_tra, 2))
```

Figura 3.3. Los 10 productos con mayor frecuencia en las transacciones (empleando ggplot2)



Fuente: elaboración propia.

```
##     items                                  transactionID
## [1] {BLUE POLKADOT WRAP,
##      CAST IRON HOOK GARDEN FORK,
##      CHILDRENS APRON APPLES DESIGN,
##      COFFEE MUG APPLES DESIGN,
##      COFFEE MUG PEARS  DESIGN,
##      JAM MAKING SET PRINTED,
##      JUMBO BAG PINK POLKADOT,
##      JUMBO BAG RED RETROSPOT,
##      LOVE HEART NAPKIN BOX ,
##      PEG BAG APPLES DESIGN,
##      RECIPE BOX RETROSPOT ,
##      RECYCLING BAG RETROSPOT ,
##      RED RETROSPOT CHILDRENS UMBRELLA,
##      RED RETROSPOT SHOPPER BAG,
##      RED RETROSPOT WRAP ,
##      SET OF 6 T-LIGHTS EASTER CHICKS,
##      WHITE HANGING HEART T-LIGHT HOLDER}        539993
## [2] {}                                          539997
```

Veamos ahora las 3 últimas.

```
# Inspeccionar las tres últimas transacciones
inspect(tail(datos_tra, 3))
```

```
##     items                                   transactionID
## [1] {COFFEE SCENT PILLAR CANDLE,
##      DAIRY MAID TOASTRACK,
##      GROW A FLYTRAP OR SUNFLOWER IN TIN,
##      ROSES REGENCY TEACUP AND SAUCER }           548547
## [2] {LOVE LARGE WOOD LETTERS ,
##      REGENCY CAKESTAND 3 TIER}                  548548
## [3] {CAKE PLATE LOVEBIRD WHITE,
##      SWEETHEART CAKESTAND 3 TIER,
##      ZINC FINISH 15CM PLANTER POTS}              548549
```

Y veamos la canasta 102.

```
# Inspeccionar la transacciones 102
inspect(datos_tra[102])
```

```
##     items                                  transactionID
## [1] {DOLLY HONEYCOMB GARLAND,
##      GUMBALL COAT RACK,
##      MINI FUNKY DESIGN TAPES,
##      STRAWBERRY HONEYCOMB  GARLAND ,
##      TRADITIONAL WOODEN CATCH CUP GAME }        540263
```

3.4 Construcción de las reglas

Como lo hemos discutido, el objetivo principal del **MBA** es encontrar reglas de asociación que permitan tomar decisiones. También discutimos en el Capítulo 2 cómo el algoritmo *Apriori* nos puede ayudar a encontrar relativamente rápido reglas que tengan sentido. A continuación implementaremos el algoritmo *Apriori* con el paquete *arules* empleando la función **apriori()**.

Para calcular las reglas de asociación empleando el algoritmo *Apriori*, esta función solo necesita dos argumentos. Primero, los datos de la clase **transactions**. El segundo argumento corresponde a los parámetros para la búsqueda. Los parámetros permiten emplear una lista para establecer el umbral mínimo de soporte (**supp**), de confianza (**conf**) y el número mínimo de ítems en la regla (**minlen**). Para el caso del umbral del soporte, el valor por defecto es 0.1 (10 %); para la confianza, el valor mínimo por defecto es 0.8 (80 %). Recuerda que si no se especifican estos umbrales, se emplearán los valores por defecto. Finalmente, por defecto, el número mínimo de ítems en las reglas encontradas es 1 (**minlen = 1**). Esto implica que el algoritmo podrá encontrar reglas en las que el antecedente (a la izquierda **LHS**) sea un *itemset* vacío y el consecuente (resultado o a la mano derecha **RHS**) un elemento. Es decir, reglas como $\{\} \rightarrow \{B\}$ o $\emptyset \rightarrow \{B\}$. Esta regla implica que se compraría B dado que aún la canasta está vacía. Normalmente, este tipo de reglas no son muy interesantes y, por tanto, es mejor emplear **minlen = 2** para no permitir reglas con conjuntos vacíos en el antecedente.

Encontremos las reglas de asociación con el algoritmo *Apriori* para las transacciones de la empresa en línea con un soporte y confianza mínimos del 1 %[8] y 80 %, respectivamente. Esto se puede hacer empleando el siguiente código:

```
# Aplicar el algoritmo Apriori
reglas <- apriori(datos_tra, parameter = list(supp = 0.01, conf = 0.8, minlen =
↪  2))
# Chequear el tipo de objeto
class(reglas)

# Mostrar las reglas
reglas
```

Hemos encontrado 324 reglas que están almacenadas en el objeto `reglas`. Estas reglas no están guardadas en un orden en especial. Por eso es importante organizarlas antes de verlas, concentrémonos por ahora solo en las cinco primeras reglas. Organicemos las reglas por *lift* e inspecciones las cinco primeras reglas empleando el siguiente código:

```
# inspeccionar las 5 primeras reglas con mayor lift
inspect(head(sort(reglas, by = "lift"), 5))
```

[8] Este soporte parece muy pequeño, pero recuerda que los cuatro ítems con la mayor frecuencia relativa están muy cerca a 0.1. Si empleamos un soporte igual a 0.10, terminamos sin reglas (ver Figura 3.3). ¡Inténtalo! Por eso, empleamos un soporte relativamente bajo. Al final de la Sección 3.5.1 se discutirá más en detalle esta elección del soporte.

Los resultados, por ser muy grandes, no se muestran en la versión pdf del libro. Si deseas comparar tus resultados con los nuestros, los encontrarás en la versión html del libro (http://www.icesi.edu.co/editorial/intro-analisis-canastas-web/).

De manera similar, inspeccionemos las cinco reglas con la mayor confianza empleando la siguiente línea de código:

```
# inspeccionar las 5 primeras reglas con mayor confianza
inspect(head(sort(reglas, by = "confidence"), 5))
```

Los resultados, por ser muy grandes, no se muestran en la versión pdf del libro. Si deseas comparar tus resultados con los nuestros, los encontrarás en la versión html del libro (http://www.icesi.edu.co/editorial/intro-analisis-canastas-web/).

Y finalmente, miremos las cinco reglas con el mayor soporte empleando el siguiente código:

```
inspect(head(sort(reglas, by = "support"), 5))
```

Los resultados, por ser muy grandes, no se muestran en la versión pdf del libro. Si deseas comparar tus resultados con los nuestros, los encontrarás en la versión html del libro (http://www.icesi.edu.co/editorial/intro-analisis-canastas-web/).

Nota que, dependiendo del criterio que empleemos, las reglas más importantes son diferentes.

3.5 Trabajando con las reglas

Hasta el momento hemos encontrado 324 reglas. Estas son muchas reglas para analizar, esto es común cuando se hace un **MBA**. Como lo discutimos en la Sección 2.6, algunas reglas pueden ser redundantes, en el sentido de que no aportan conocimientos adicionales. Recordemos que una regla es **redundante** si existe otra regla más general con una confianza igual o mayor. Una regla más general es aquella que tiene el mismo *itemset* como consecuente (**RHS**), pero uno o más elementos menos en el antecedente (**LHS**) (Ver Sección 2.6).

Así, una de las primeras tareas que tenemos que hacer con las reglas encontradas es descartar aquellas que sean redundantes. Las reglas redundantes se pueden encontrar con la función **is.redundant()** del paquete *arules*. Para nuestro ejemplo, las reglas redundantes las podemos encontrar de la siguiente manera:

```
# Encontrar reglas redundantes
reglas_redundantes <- is.redundant(reglas)
# Chequear clase del objeto
class(reglas_redundantes)
```

```
## [1] "logical"
```

```
# Contar el número de reglas redundantes
sum(reglas_redundantes)
```

```
## [1] 30
```

Tenemos 30 reglas redundantes. Podemos descartar esas reglas redundantes y quedarnos con aquellas que sí son pertinentes de la siguiente manera:

```
# Filtrar las reglas redundantes y mantener solo las no redundantes
reglas <- reglas[!reglas_redundantes]
```

Ahora contamos con 294 reglas (no redundantes). Si queremos, podemos transformar las reglas a un objeto de clase **data.frame** con el siguiente código:

```
# Transformar reglas a clase data.frame
reglas_df = as(reglas, "data.frame")
# visualizar las reglas no redundantes
glimpse(reglas_df)
```

```
## Rows: 294
## Columns: 6
## $ rules      <chr> "{HERB MARKER CHIVES } => {HERB MARKER BASIL}", "{HERB MARK~
## $ support    <dbl> 0.01006089, 0.01059042, 0.01085518, 0.01138470, 0.01111994,~
## $ confidence <dbl> 0.8444444, 0.8888889, 0.9111111, 0.9555556, 0.9333333, 0.88~
## $ coverage   <dbl> 0.01191422, 0.01191422, 0.01191422, 0.01191422, 0.01191422,~
## $ lift       <dbl> 61.33590, 63.34591, 60.37310, 62.22644, 60.77931, 58.90058,~
## $ count      <int> 38, 40, 41, 43, 42, 40, 38, 42, 43, 43, 47, 47, 48, 48, 47,~
```

Esto permitiría manipular las reglas como estamos acostumbrados. Por ejemplo, podrías emplear los paquetes *dplyr* (Wickham et al., 2021) o *ggplot2* para seguir analizando las reglas.

3.5.1 Jugando con los parámetros del algoritmo Apriori (avanzado)

Seleccionar el umbral mínimo para el soporte y la confianza afecta sustancialmente el número de reglas con las que finalmente terminamos. Escoger estos umbrales (parámetros) para el algoritmo no es una tarea sencilla. La elección de los umbrales de soporte y confianza es un verdadero problema práctico al emplear los algoritmos de minería de reglas de asociación. Por ejemplo, buscar reglas de asociación empleando un umbral de soporte alto elimina las reglas con ítems inusuales sin tener en cuenta el valor de confianza de estas reglas. Por otra parte, cuando el umbral de soporte es bajo, se genera un elevado número de reglas y, en consecuencia, al usuario final le resulta muy difícil, si no imposible, utilizarlas. En otras palabras, es importante jugar un poco con estos parámetros hasta encontrar el resultado adecuado para cada juego de datos.

En algunos casos nos podemos ayudar un poco de la fuerza "bruta" de los computadores para entender el efecto final sobre el número de reglas de asociación de nuestra elección de dichos umbrales.

Calculemos el número de reglas de asociación que encontrará el algoritmo *Apriori*

con un soporte de 0.01. Para esto podemos emplear *loops*[9] de la siguiente manera (asegúrate que puedes seguir el código):

```
# Fijar los niveles para la confianza
confidenceLevels <- seq(from = 0.95, to = 0.5, by = -0.05)

# Vector vacío para guardar los resultados
r_res_01 <- NULL

# Loops para Algoritmo Apriori con soporte del 0.01
for (i in 1:length(confidenceLevels)) {
    r_res_01[i] <- length(apriori(datos_tra, parameter = list(sup = 0.01, conf
↪  = confidenceLevels[i]),
        control = list(verbose = F)))
}
```

Ahora grafiquemos estos resultados. El siguiente código genera la Figura 3.4:

```
# Construir el data.frame con datos a visualizar
data_loops <- as.data.frame(confidenceLevels, r_res_01)
# Visualizar los resultados
ggplot(data_loops, aes(x = confidenceLevels, y = r_res_01)) + geom_point() +
↪  geom_line() +
    xlab("Confianza") + ylab("Número de reglas encontradas") + theme_minimal()
```

Podemos ver cómo se afecta nuestro número de reglas cuando el soporte aumenta. Y comparemos este resultado con un soporte más alto (0.02).

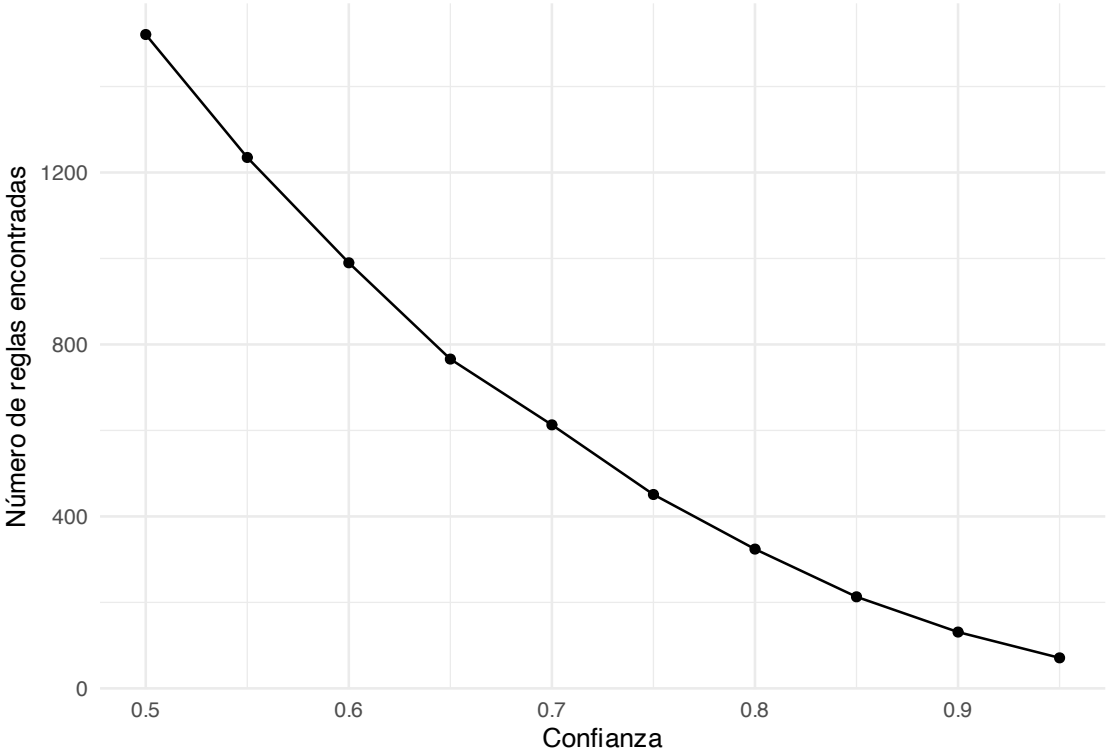
```
# Vector vacío para guardar los resultados
r_res_02 <- NULL

# Loops para Algoritmo Apriori con soporte del 0.01
for (i in 1:length(confidenceLevels)) {
    r_res_02[i] <- length(apriori(datos_tra, parameter = list(sup = 0.02, conf
↪  = confidenceLevels[i]),
        control = list(verbose = F)))
}


# Unir los resultados
nb_rules = data.frame(r_res_01, r_res_02, confidenceLevels)
```

Y finalmente, visualicemos estos resultados (Ver Figura 3.5).

Esto justifica por qué nuestro soporte fue tan bajo para este caso.

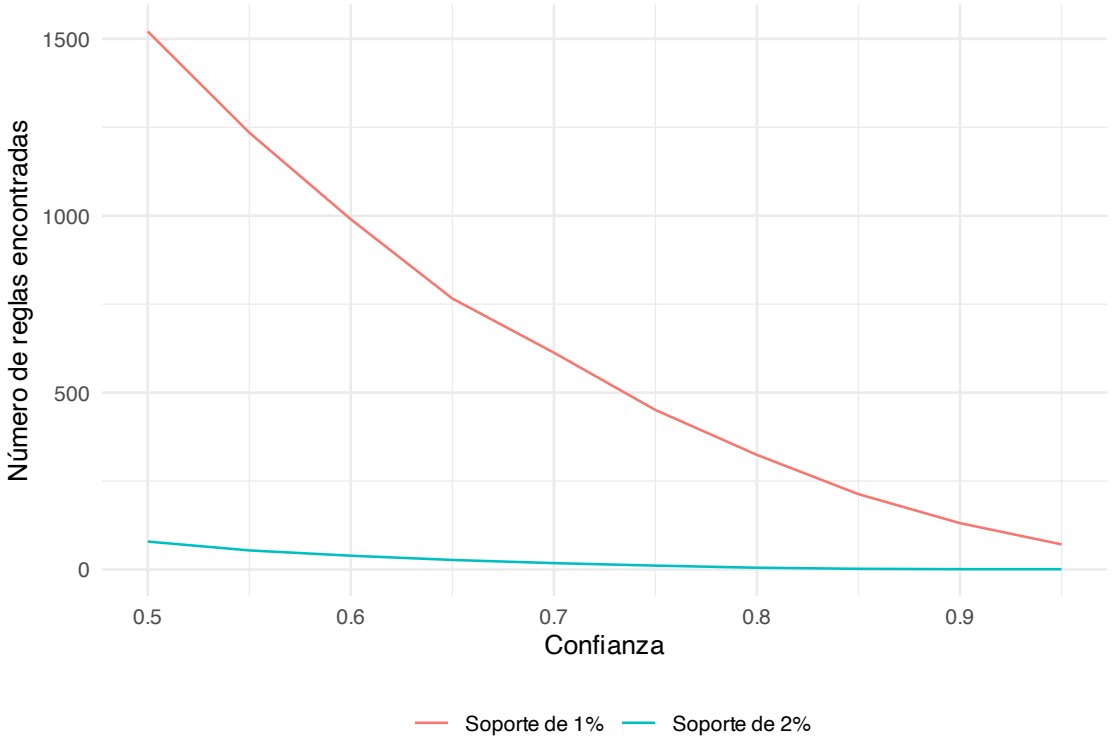
[9] Para una introducción a los *loops*, puedes consultar Alonso (2021).

Figura 3.4. Número de reglas encontradas por el algoritmo Apriori para diferentes valores de confianza manteniendo el soporte en 0.01



Fuente: elaboración propia.

Figura 3.5. Número de reglas encontradas por el algoritmo Apriori para diferentes valores de confianza con soportes de 0.01 y 0.02



Fuente: elaboración propia.

3.5.2 Trabajando con reglas para productos específicos

En algunas ocasiones el tomador de decisiones estará interesado en un producto en específico. Por ejemplo, se desea emplear las reglas de asociación para promover la venta de un producto. En este caso, nuestro producto deberá estar en el consecuente (**LHS**) de la regla. O podemos querer buscar un producto que pueda ser impulsado en forma de combo con el producto estrella. En ese caso nos interesará buscar reglas con el producto estrella como antecedente (**RHS**) de la regla.

Así, es frecuente que en la práctica queramos buscar *itemsets* específicos o reglas específicas dentro del conjunto de todas las reglas no redundantes extraídas. Esto se puede hacer de manera sencilla en R. La función **apriori()** tiene otro argumento que aún no hemos usado, que es **appearance** (una traducción en este contexto sería aparición). Con este argumento podemos encontrar aquellas reglas que tengan como **LHS** o **RHS** los *itemsets* deseados.

Supongamos por un momento que queremos encontrar las reglas de asociación que permitan impulsar la compra del producto "HERB MARKER THYME"[10]. Es decir, reglas que tengan al producto "HERB MARKER THYME" a la derecha de la regla (**RHS**). Esto lo podemos hacer empleando el argumento **appearance**, agregando como lista el requerimiento que deseamos; en este caso **rhs = "HERB MARKER THYME"**. Es decir, el código sería el siguiente:

```
# Aplicar el algoritmo Apriori para encontrar reglas con 'HERB MARKER THYME'
↪  en
# el rhs
regla_thyme_rhs <- apriori(datos_tra, parameter = list(supp = 0.01, conf = 0.8,
↪  minlen = 2),
    appearance = list(rhs = "HERB MARKER THYME"), control = list(verbose = F))

regla_thyme_rhs
```

```
## set of 22 rules
```

Nota que además incluimos el argumento **control = list(verbose=F)**. Esto evita que se impriman los resultados en la consola de mensajes mientras se realiza el cálculo. Encontramos 22 reglas con este ítem en el consecuente. Igual que antes, podemos descartar las reglas redundantes y visualizar los resultados ordenando por alguna de las métricas de interés.

```
# Encontrar reglas redundantes
regla_thyme_rhs_red <- is.redundant(regla_thyme_rhs)
# Contar el número de reglas redundantes
sum(regla_thyme_rhs_red)
```

```
## [1] 8
```

[10] Un "HERB MARKER" (marcador de hierbas) es una pequeña estaca o etiqueta, usualmente de madera, plástico o cerámica resistente a la intemperie, que se clava en la matera o la huerta para identificar la especie aromática sembrada. Facilita el cuidado del cultivo y la correcta recolección de cada hierba.

```
# Crear objeto con reglas no redundantes
regla_thyme_rhs = regla_thyme_rhs[!regla_thyme_rhs_red]
# Ver el objeto con las reglas no redundantes
regla_thyme_rhs
```

```
## set of 14 rules
```

Ahora miremos las cinco reglas con la mayor confianza. Para esto podemos emplear el siguiente código:

```
# Inspeccionar las top 5 reglas por confianza
inspect(head(sort(regla_thyme_rhs, by = "confidence"), 5))
```

Los resultados, por ser muy grandes, no se muestran en la versión pdf del libro. Si deseas comparar tus resultados con los nuestros, los encontrarás en la versión html del libro (http://www.icesi.edu.co/editorial/intro-analisis-canastas-web/).

¿Qué puedes concluir? ¿Qué acción de mercadeo puedes sugerir?

Ahora miremos reglas que tengan como antecedente (**LHS**) el ítem "HERB MARKER ROSEMARY". De manera análoga, podemos emplear el siguiente código:

```
regla_rosemary_lhs <- apriori(datos_tra, parameter = list(supp = 0.01, conf =
↪  0.8,
    minlen = 2), appearance = list(lhs = "HERB MARKER ROSEMARY"), control =
    ↪  list(verbose = F))

# Encontrar reglas redundantes
regla_rosemary_lhs_red <- is.redundant(regla_rosemary_lhs)
# Contar el número de reglas redundantes
sum(regla_rosemary_lhs_red)
```

```
## [1] 0
```

```
# no hay reglas redundantes

# Inspeccionar las top 5 reglas por confianza
inspect(sort(regla_rosemary_lhs, by = "confidence"))
```

```
##     lhs                          rhs                    support    confidence
## [1] {HERB MARKER ROSEMARY} => {HERB MARKER THYME}   0.01456182 0.9649123
## [2] {HERB MARKER ROSEMARY} => {HERB MARKER MINT}    0.01350278 0.8947368
## [3] {HERB MARKER ROSEMARY} => {HERB MARKER BASIL}   0.01244374 0.8245614
## [4] {HERB MARKER ROSEMARY} => {HERB MARKER PARSLEY} 0.01244374 0.8245614
##     coverage   lift     count
## [1] 0.01509134 62.83575 55
## [2] 0.01509134 58.26588 51
## [3] 0.01509134 59.89170 47
## [4] 0.01509134 58.76167 47
```

En este resultado se ha definido la etiqueta (*herb marker*) para romero (*Rosemery*) como el antecedente de la regla de asociación. Por lo tanto, el consecuente nos indica los productos que los consumidores compran dado que ya tienen en sus carritos de compra la etiqueta para romero. En este caso, las cuatro reglas encontradas tienen un *lift* muy grande (superior a 1) y una confianza mayor a 82.5%. Es decir, se trata de reglas con una alta probabilidad de ocurrencia. Los productos que se encuentran como consecuente, cuando el antecedente es la etiqueta para romero, son las etiquetas para el tomillo (*thyme*), la menta (*mint*), la albahaca (*basil*) y el perejil (*parsley*).

El soporte de cada una de estas reglas (frecuencia con la que un consumidor compra la cesta que contiene la etiqueta para el romero y las otras etiquetas de hierbas) es de aproximadamente el 1%. Adicionalmente, una vez el consumidor incluye la etiqueta para romero en su canasta de compras, hay una confianza del 96% de que también incluya la etiqueta para tomillo, del 82% de que incluya la etiqueta para la menta y del 82% de que incluya la etiqueta para el perejil.

Teniendo en cuenta el resultado del *lift* (la popularidad conjunta de ambos ítems), vemos que hay una asociación positiva (*lift* > 1) entre la compra de la etiqueta para romero y la compra de cada una de las otras etiquetas de hierbas (tomillo, menta, albahaca o perejil). La cobertura (*coverage*) nos indica que la regla etiqueta de romero-tomillo (o una de las otras etiquetas para hierbas) se aplica con una probabilidad del 1.5%.

La pregunta que te puedes estar haciendo en estos momentos es: ¿por qué quisiera el tomador de decisiones observar las reglas de asociación con el antecedente de etiquetas para romero? Al encontrar estas reglas (reglas con el ítem "HERB MARKER ROSEMARY" en el **LHS**), por ejemplo, sería posible observar cuáles productos podrían ofrecerse de manera conjunta con la etiqueta de romero para generar valor al consumidor.

El combo es relevante para el consumidor porque le ofrece un producto complementario de su interés; adicionalmente, obtiene ambos productos a un menor precio en comparación con la compra individual.

3.6 Comentarios finales

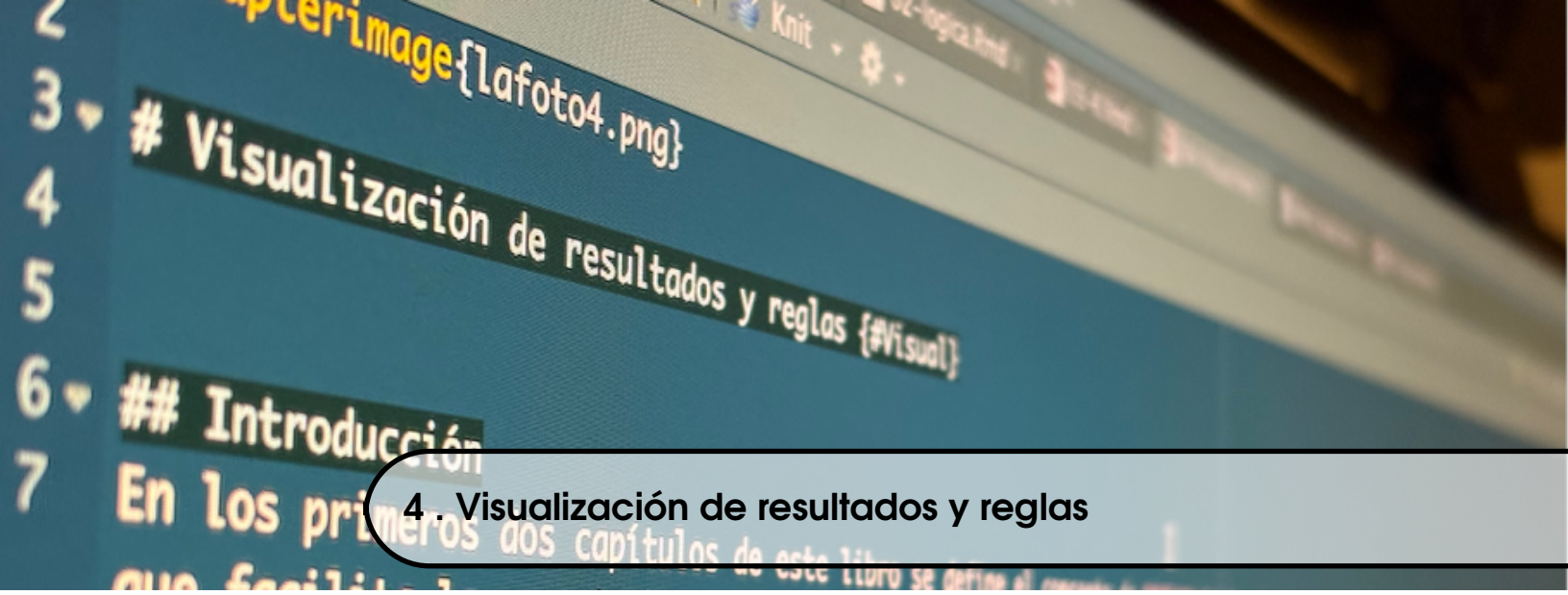
En este Capítulo hemos estudiado paso a paso cómo realizar un **MBA** en R. Empleando una base relativamente grande con datos transaccionales, pudimos encontrar reglas de asociación no redundantes. Estas reglas de asociación son el insumo para la toma de decisiones en el negocio. De esta manera, tanto el científico de datos como el *analytics translator* deben conocer y establecer conjuntamente las reglas de asociación con las que se alimenta el **MBA**; este es el resultado que guiará las decisiones estratégicas. Aunque el científico de datos se encarga del detalle técnico en R, es importante que el *analytics translator* comprenda el proceso de creación de reglas para entender las implicaciones de las reglas establecidas. De estas reglas depende el resultado y los *insights* que conllevan a la toma de decisión.

Una vez que contamos con las reglas de asociación, las preguntas de negocio em-

pezarán a fluir y la tarea del científico de datos termina e inicia la del **analytics translator** para acompañar en el uso de dichas reglas para la toma de decisiones. En el Capítulo 4 estudiaremos cómo visualizar los resultados para facilitar este proceso de toma de decisiones.

Antes de terminar este Capítulo, guarda el espacio de trabajo (*workspace*), lo emplearemos en el Capítulo 4. Eso lo puedes hacer con el siguiente código:

```
save.image(file = "I_A_C_C3.RData")
```



4 . Visualización de resultados y reglas

4.1 Introducción

En los primeros dos capítulos de este libro se define el concepto de **MBA** (sigla del término en inglés *Market Basket Analysis* o análisis de canasta) junto con un ejemplo que facilita la apropiación del concepto. En el Capítulo 3 estudiamos cómo aplicar los conceptos empleando R para encontrar reglas de asociación que cumplieran unos criterios mínimos de soporte y confianza empleando el algoritmo *Apriori*. Adicionalmente, vimos cómo extraer reglas de asociación para productos específicos, ya fueran como antecedentes en la regla (**LHS**) o consecuente (**RHS**).

En este Capítulo nos concentraremos en cómo visualizar los resultados y cómo brindar herramientas interactivas a los tomadores de decisiones a partir de las reglas que aprendimos a encontrar en el Capítulo 3. La tarea de visualizar los resultados del **MBA** es una tarea compartida entre científico de datos y **analytics translator**. El **analytics translator** garantizará que las visualizaciones producidas por los científicos de datos sean de fácil comprensión para los tomadores de decisiones de la organización.

En el trabajo del científico de datos y del **analytics translator**, es importante, después de encontrar las reglas de asociación, construir visualizaciones que transmitan confianza en el análisis realizado y que también permitan usar los resultados para tomar decisiones. En ese orden de ideas, este Capítulo estará centrado en dos tipos de visualizaciones: aquellas que comunican las métricas de las reglas de asociación (Ver Sección 4.2) y aquellas que permiten ver las reglas de asociación e interactuar con ellas (Ver Sección 4.3).

Este Capítulo explica al científico de datos los pasos para generar las visualizaciones y al *analytics translator* le permitirá conocer el tipo de visualizaciones que se emplean en este contexto y cómo se interpretan. El *analytics translator* puede omitir de este Capítulo el detalle del código en R. Lo que sí será importante para este rol es tener claras las herramientas disponibles para comunicar los resultados y su interpretación. Además, siempre es importante tener en cuenta que la construcción de visualizaciones es un

trabajo conjunto entre el científico de datos y el *analytics translator* para asegurarse de que los *insights* son comunicados adecuadamente a los tomadores de decisiones.

En este Capítulo continuaremos con el ejemplo del capítulo anterior; carguemos el espacio de trabajo (*workspace*) que creamos al final del Capítulo 3. Esto lo puedes hacer con el siguiente código:

```
load(file = "I_A_C_C3.RData")
```

Recordemos que tenemos en el espacio de trabajo (*workspace*) por lo menos los siguientes objetos:

- `datos_tra`: objeto de clase **transactions** con los datos de todas las transacciones de una empresa de comercio electrónico con sede en el Reino Unido para el primer trimestre de un año reciente.
- `Datos_original`: objeto de clase **data.frame** que contiene todas las variables originales.
- `reglas`: reglas de asociación detectadas con el algoritmo *Apriori* (umbral para el soporte de 0.01 y para la confianza de 0.8).
- `regla_thyme_rhs`: reglas de asociación detectadas con el algoritmo *Apriori* (umbral para el soporte de 0.01 y para la confianza de 0.8) con el ítem "HERB MARKER THYME" como consecuente (**RHS**).
- `regla_rosemary_lhs`: reglas de asociación detectadas con el algoritmo *Apriori* (umbral para el soporte de 0.01 y para la confianza de 0.8) con el ítem "HERB MARKER ROSEMARY" como antecedente (**LHS**).

En todos los casos, las reglas de asociación redundantes fueron descartadas. A continuación, estudiaremos algunas de las opciones disponibles para visualizar las métricas de los *itemsets* y de las reglas de asociación, así como las reglas de asociación como tal.

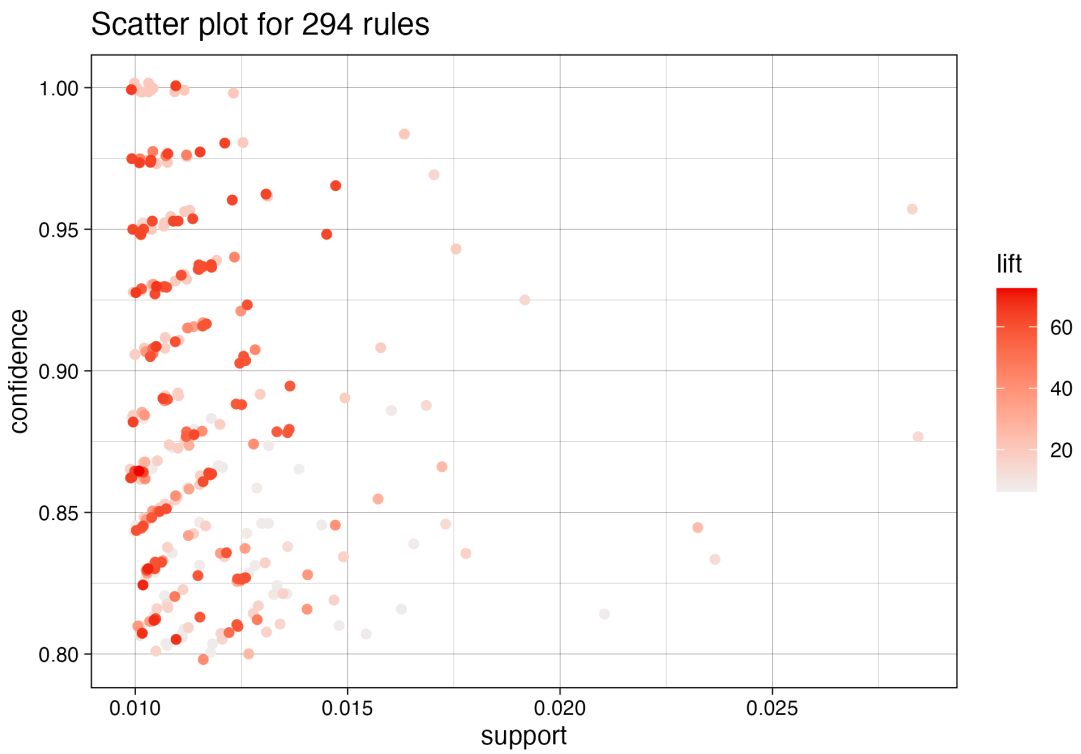
4.2 Visualizando las métricas de las reglas de asociación

Una vez que se ha empleado el algoritmo *Apriori* para elegir reglas de asociación, es común que se quiera visualizar las métricas de las reglas seleccionadas. Un gráfico común es un gráfico de dispersión con el soporte y la confianza. El paquete *arulesViz* (Hahsler, 2017) provee varias opciones para visualizar las métricas de las reglas y las reglas como tal. Este paquete incluye la función **plot()** para objetos de clase **rules**[1]. Esta función por defecto presenta una visualización del soporte (eje horizontal), la confianza (eje vertical) y el *lift* (el color). Por ejemplo, la Figura 4.1 permite visualizar estas tres métricas de cada una de las 294 reglas no redundantes encontradas en nuestro ejemplo.

En la Figura 4.1 podemos ver tres *insights* importantes para la toma de decisiones. Primero, el soporte de la mayoría de las 294 reglas se concentra alrededor del 1 %. Las

[1] Recuerda que cuando tenemos funciones que tienen el mismo nombre en diferentes paquetes, típicamente estas están diseñadas para reaccionar distinto de acuerdo a la clase del objeto que se use en el argumento.

Figura 4.1. Soporte, confianza y lift de todas las reglas no redundantes encontradas por el algoritmo Apriori



Fuente: elaboración propia.

reglas que descubre el algoritmo *Apriori* aparecen en muy pocas tirillas de compra (carritos de compra o tickets). Esto indica nichos específicos de comportamiento, con poco volumen, pero muy definidos, que podrían ser explotados con tácticas "*one-to-one*" (recomendaciones personalizadas, ofertas en la app, *displays* en punto de venta dirigidos a microsegmentos). Segundo, la confianza es sistemáticamente alta (entre 0.8 y 1.0) y el color rojo intenso señala *lifts* superiores a 30, incluso por encima de 60. En otras palabras, cuando el antecedente está presente, el consecuente casi siempre se añade al carrito y la probabilidad conjunta supera decenas de veces la esperada por azar. Esto valida la posibilidad de crear combos (*bundles*) o descuentos cruzados. Activar el producto "gatillo" garantiza con una alta probabilidad la venta del complementario (consecuente). Tercero, existen pocas reglas con soportes cercanos a la zona entre el 2.0 % y el 2.5 % y confianza mayor 0.85. Si bien el *lift* es algo menor, cubren una base de clientes mayor y, por tanto, son candidatos idóneos para campañas masivas (promociones en góndola, *emailing* general). Así, el gráfico sugiere un portafolio de reglas dual; por un lado, reglas de alto *lift*-bajo soporte para acciones hipersegmentadas que impulsen ticket promedio y fidelización, y por el otro lado, reglas de soporte medio-alto para iniciativas de *category management* que muevan volumen sin sacrificar la rentabilidad.

El código que generó Figura 4.1 es el siguiente:

```
# Cargar el paquete
library(arulesViz)
plot(reglas)
```

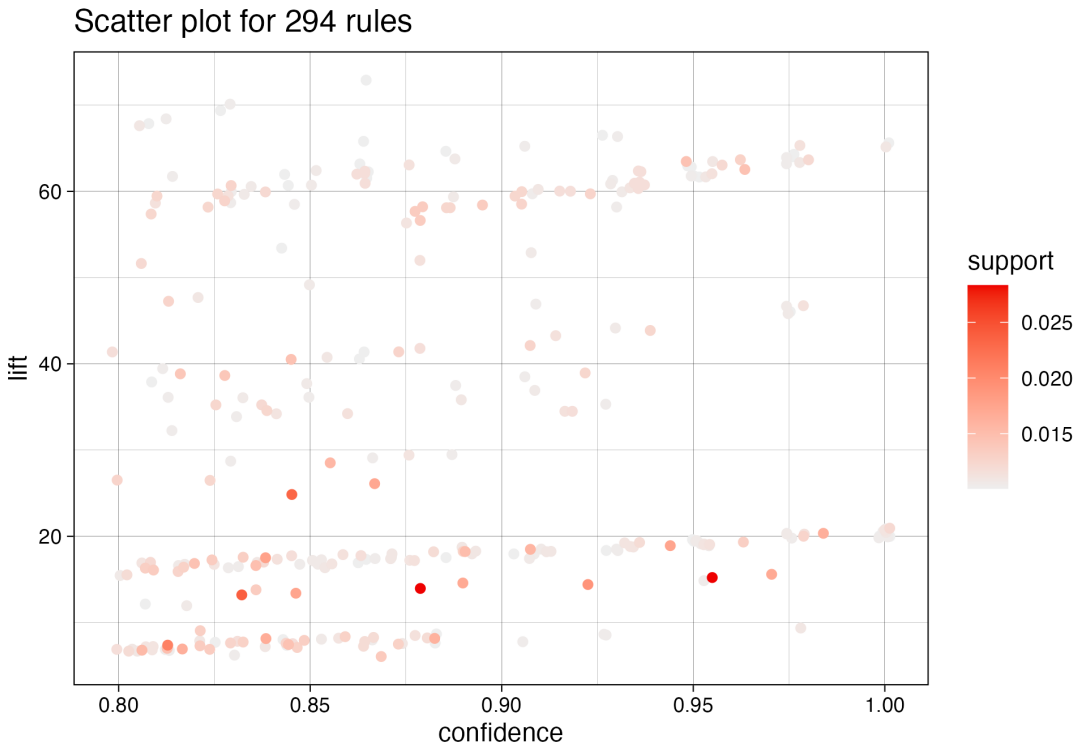
La función **plot()** del paquete *arulesViz* permite cambiar las métricas que se presentan en los dos ejes empleando el argumento **measure**. Con el argumento **shading** podemos cambiar la métrica que le da color a los puntos. Por ejemplo, la Figura 4.2 se construye con el siguiente código:

```
# Visualizar las métricas de las reglas
plot(reglas, measure = c("confidence", "lift"), shading = "support")
```

En la Figura 4.2 invertimos la perspectiva: el *lift* pasa al eje vertical y el soporte se codifica con la intensidad del color. Esta visualización alternativa refuerza los hallazgos ya descritos para la Figura 4.1. Los puntos más oscuros, reglas con mayor soporte, se agrupan en bandas de *lift* moderado (entre 15 y 25) y confianza alta (mayor a 0.85), confirmando su potencial para campañas masivas de *category management*. A la vez, se distinguen nubes más claras en la zona superior del gráfico (*lift* mayor a 50 y soporte menor al 1 %), evidencia de oportunidades hipersegmentadas donde, aun con poco volumen, la probabilidad de compra conjunta se multiplica. Exactamente el tipo de reglas que habíamos recomendado para tácticas "*one-to-one*". Así, ambas visualizaciones permiten comunicar un mismo mensaje estratégico: gestionar el portafolio de reglas según la combinación soporte-confianza-*lift* para equilibrar volumen y rentabilidad.

Adicionalmente, tenemos la opción de visualizar las métricas de manera diferente con el argumento **method**. Por defecto, la función **plot()** crea un gráfico de disper-

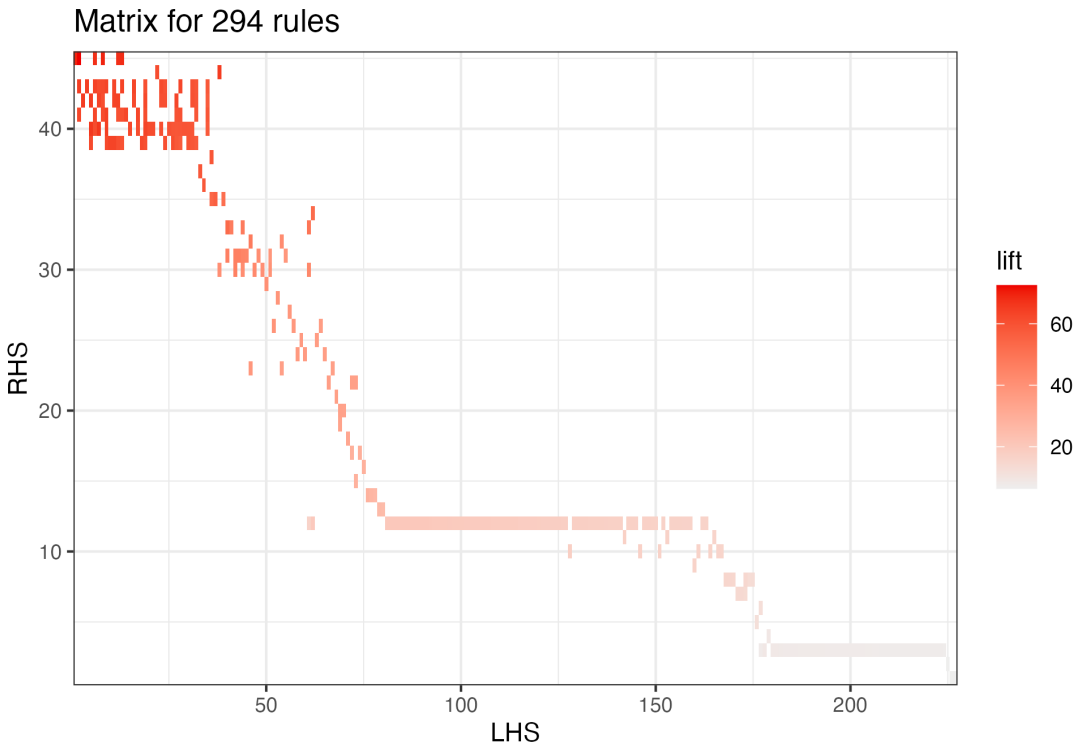
Figura 4.2. Soporte, confianza y lift de todas las reglas no redundantes encontradas por el algoritmo Apriori (Versión 2)



Fuente: elaboración propia.

sión (**method = "scatterplot"**). Con este argumento podemos crear gráficos en forma de matriz (**method = "matrix"**) o en tres dimensiones[2] **method = "matrix3D"**. En las Figuras 4.3 y 4.4 se reportan las métricas de todas las reglas empleando dos diferentes visualizaciones. ¡Intenta reproducir esas figuras!

Figura 4.3. Soporte, confianza y lift de todas las reglas no redundantes encontradas por el algoritmo Apriori (Versión matriz)



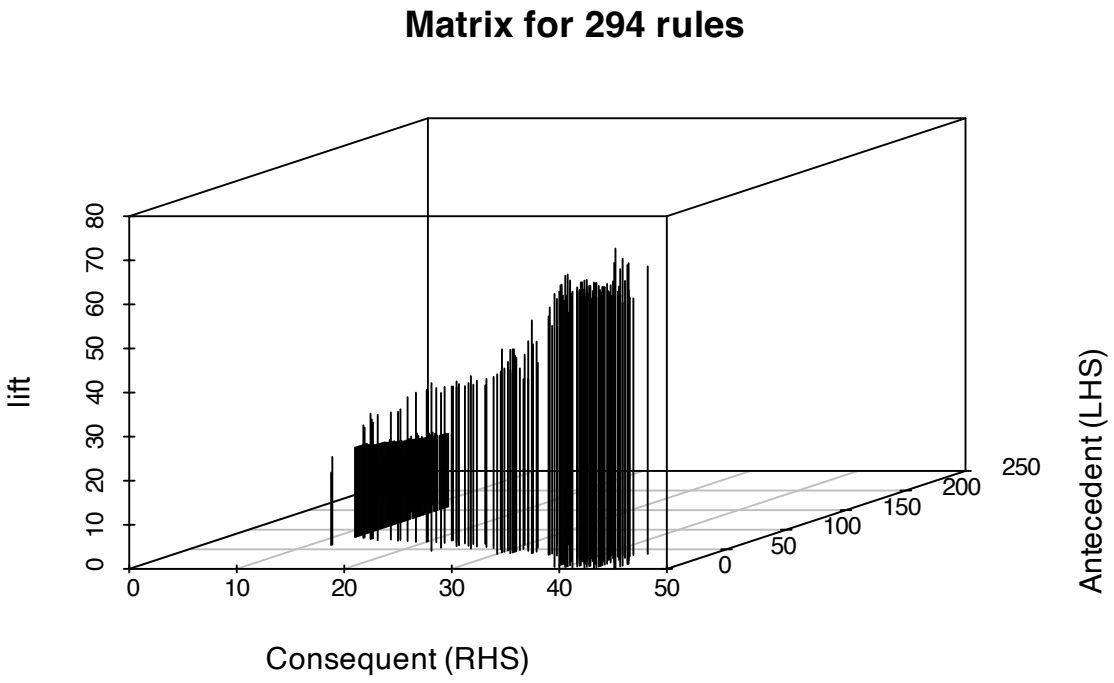
Fuente: elaboración propia.

En la Figura 4.3, obtenida con **method = "matrix"**, cada celda ubica una regla según el índice de su antecedente (eje **LHS**) y de su consecuente (eje **RHS**). El tono rojizo representa el *lift*. Dos patrones sobresalen. Primero, un bloque densamente rojo en la esquina superior izquierda, señal de que un grupo de productos actúa repetidamente como "gatillo" y "complemento" con *lift* superior a 40. Y segundo, un gradiente que se reduce en diagonal a medida que los índices aumentan, mostrando que las combinaciones menos comunes mantienen un *lift* importante, aunque menor, a medida que aumenta la variedad de ítems involucrados. Esta distribución confirma que la mayor tracción comercial se concentra en un subconjunto limitado de artículos clave, ideales para definir tácticas de *cross-selling* prioritarias.

La Figura 4.4 se crea con **method = "matrix3D"** y proyecta esas mismas reglas en un

[2]Este tipo de visualización no es recomendable; en la mayoría de las situaciones es muy confuso.

Figura 4.4. Soporte, confianza y lift de todas las reglas no redundantes encontradas por el algoritmo Apriori (Versión 3D)



Fuente: elaboración propia.

espacio tridimensional (**LHS**, **RHS**, *lift*). Aunque más difícil de leer, de ahí nuestra nota de cautela, la "selva" de barras verticales permite apreciar la dispersión del *lift*. La mayoría de las reglas se elevan entre 10 y 60, con unos cuantos picos que superan 70. La visualización corrobora la heterogeneidad ya detectada; pocas combinaciones generan *lifts* extraordinarios, mientras la mayoría sostiene valores sólidos que, sin llegar a ser excepcionales, respaldan estrategias de empaquetamiento estándar.

Juntas, las Figuras 4.3 y 4.4 complementan la historia narrada por las Figuras 4.1 y 4.2: un pequeño núcleo de productos concentra asociaciones fuertes y repetitivas (alto soporte y alto *lift*), alrededor del cual gravita un espectro de reglas de menor volumen pero todavía rentables.

En estos casos en los que existen tantas reglas, y que el usuario quisiera conocer las métricas de cada una de las reglas, puede ser una buena idea emplear una visualización interactiva. Las visualizaciones interactivas permiten a los usuarios realizar la exploración de datos para diferentes asociaciones. Es decir, es posible visualizar resultados para asociaciones específicas y así tomar decisiones independientes, por supuesto, utilizando el criterio del negocio.

El paquete *arulesViz* (Hahsler, 2017) también permite hacer visualizaciones interactivas. Solo requerimos emplear el argumento **engine** en la misma función **plot()**. Este argumento permite escoger qué "motor" de generación de gráficos se empleará. Ya debiste notar que los gráficos habían sido generados anteriormente con el paquete *ggplot2*. Podemos pedirle a esta función que emplee como "motor" el paquete *plotly* (Sievert, 2020). Por ejemplo, la siguiente línea de código genera la Figura 4.5:

```
# gráfico interactivo motor plotly
plot(reglas, engine = "plotly")
```

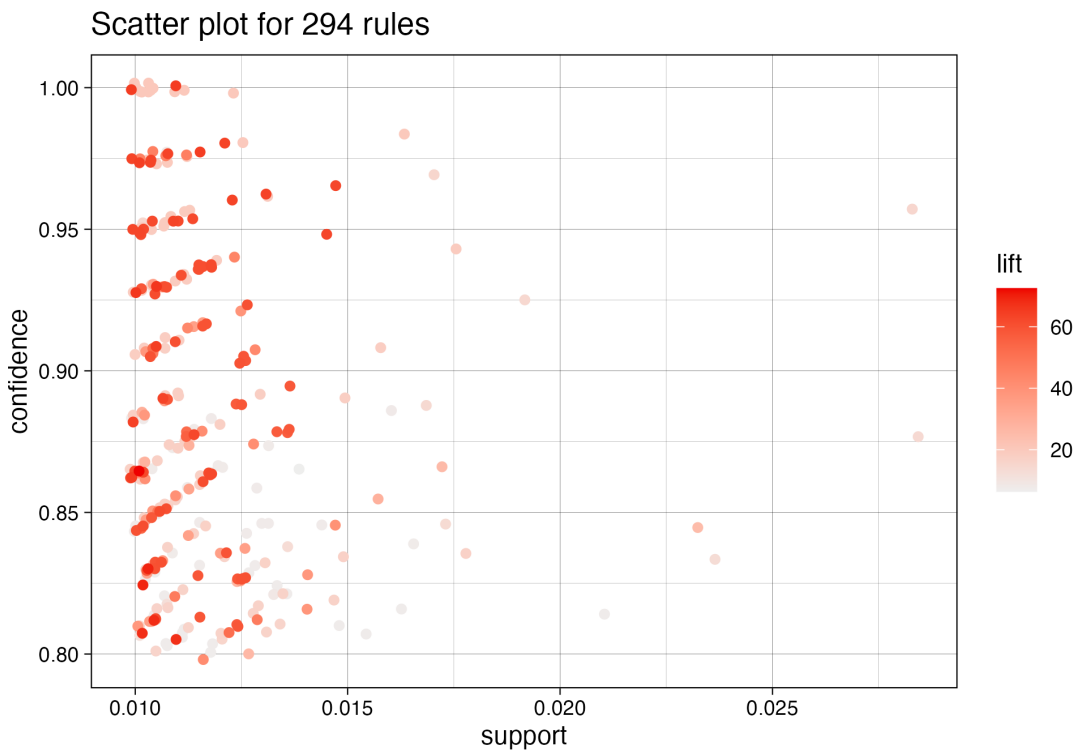
En esta versión pdf del libro la Figura 4.5 no será interactiva. Si deseas ver la versión interactiva, puedes consultar la versión html del libro (http://www.icesi.edu.co/editorial/intro-analisis-canastas-web/).

Este tipo de visualizaciones interactivas (Ver Figura 4.5) permiten al usuario interactuar con los datos. Juega un rato con este gráfico. Se pueden hacer *zoom* en partes del gráfico. Pasa el cursor por encima de un punto para ver la respectiva regla y sus métricas. La visualización interactiva solo funciona en la versión web del libro. Si deseas ver la versión interactiva, puedes consultar la versión html del libro (http://www.icesi.edu.co/editorial/intro-analisis-canastas-web/). Esta visualización permite al tomador de decisiones ver tanto la regla de asociación como cada una de las tres métricas visualizadas; de esta manera es mucho más sencillo tomar decisiones.

4.3 Visualizando las reglas

Finalmente, para el negocio, lo más importante del **MBA** son las decisiones que se puedan tomar a partir de las reglas. Por esto, más importante que conocer las métricas de las reglas será conocer las reglas en sí. Una de las opciones para visualizar las reglas son los gráficos de coordenadas paralelas.

Figura 4.5. Gráfico interactivo del soporte, confianza y lift de todas las reglas no redundantes encontradas por el algoritmo Apriori.



Fuente: elavoración propia.

Este gráfico muestra los ítems en el eje vertical y el eje horizontal representa las posiciones en una determinada regla. El gráfico emplea una flecha cuya punta muestra al artículo consecuente. Las flechas sólo abarcan la distancia horizontal que sea necesaria para representar todos los elementos de la regla (antecedentes). En otras palabras, las reglas con menos elementos son flechas más cortas. El grosor de las flechas representa el soporte y la intensidad del color representa la confianza.

Empecemos con un ejemplo sencillo. Miremos el caso de las cuatro reglas de asociación que habíamos detectado con el algoritmo *Apriori* (umbral para el soporte de 0.01 y para la confianza de 0.8) con el ítem "HERB MARKER ROSEMARY" como antecedente (**LHS**).

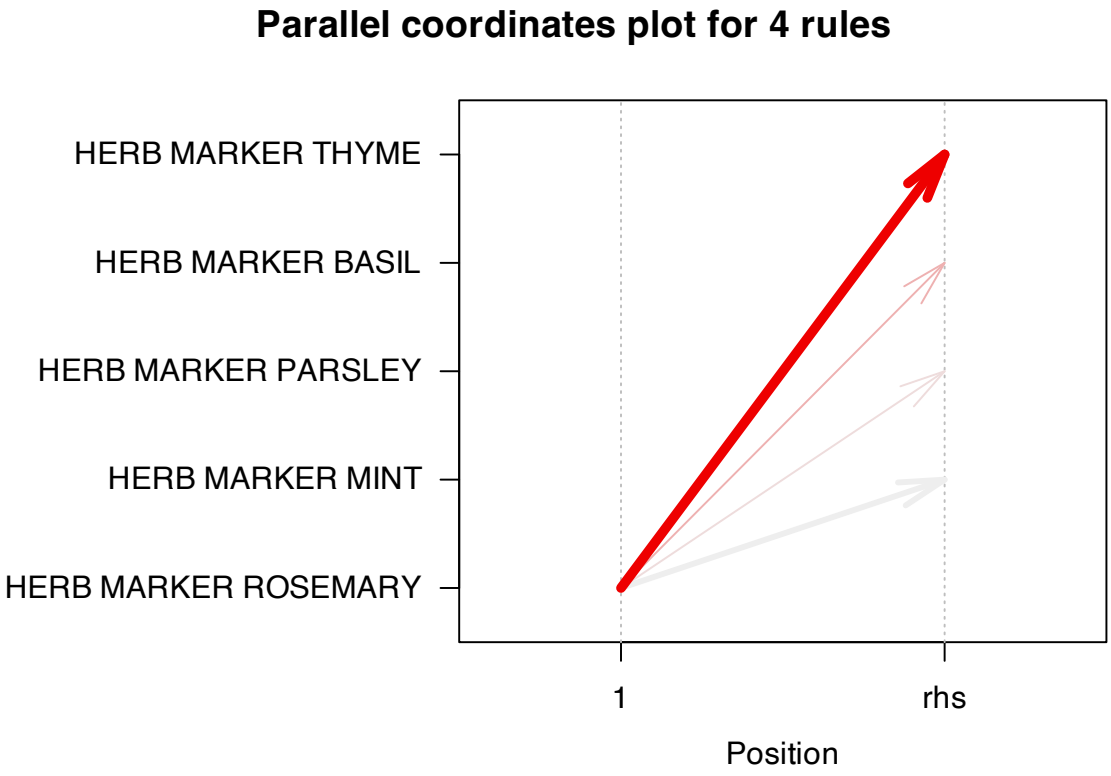
```
# inspeccionar las reglas
inspect(sort(regla_rosemary_lhs, by = "support"))
```

```
##     lhs                          rhs                    support    confidence
## [1] {HERB MARKER ROSEMARY} => {HERB MARKER THYME}   0.01456182 0.9649123
## [2] {HERB MARKER ROSEMARY} => {HERB MARKER MINT}    0.01350278 0.8947368
## [3] {HERB MARKER ROSEMARY} => {HERB MARKER BASIL}   0.01244374 0.8245614
## [4] {HERB MARKER ROSEMARY} => {HERB MARKER PARSLEY} 0.01244374 0.8245614
##     coverage   lift     count
## [1] 0.01509134 62.83575 55
## [2] 0.01509134 58.26588 51
## [3] 0.01509134 59.89170 47
## [4] 0.01509134 58.76167 47
```

Nota que estas cuatro reglas tienen por construcción como antecedente únicamente el ítem "HERB MARKER ROSEMARY" y los consecuentes son **itemsets** de un ítem. El correspondiente gráfico de coordenadas paralelas se presenta en la Figura 4.6. Podemos ver cómo la regla que tiene como consecuente (**RHS**)"HERB MARKER THYME" tiene el mayor soporte (grosor de la flecha) y confianza (color más intenso).

La Figura 4.6 traduce las cuatro reglas con antecedente "HERB MARKER ROSEMARY" en una lectura más accesible para la audiencia familiarizada con estas visualizaciones. Como se mencionó, la flecha más gruesa y de color rojo intenso conecta "ROSEMARY" con "THYME". Esta flecha indica la regla de mayor soporte y confianza dentro del conjunto. Es decir, los compradores que adquieren el marcador de romero ("HERB MARKER ROSEMARY") casi siempre añaden el de tomillo ("HERB MARKER THYME") y lo hacen con una frecuencia relativamente alta en la base de datos de transacciones estudiada. Las flechas más tenues que apuntan a los marcadores de hierbas ("HERB MARKER") de albahaca ("HERB MARKER BASIL"), perejil ("HERB MARKER PARSLEY") y menta ("HERB MARKER MINT") exhiben, respectivamente, soportes y confianzas decrecientes, lo que sugiere un vínculo comercial menos robusto, aunque todavía relevante. Para el negocio, esto implica priorizar el empaquetado o la exhibición conjunta de los marcadores de romero y tomillo; por ejemplo, un combo (*bundle*) con descuento o una ubicación contigua en góndola. Mientras que las combinaciones con albahaca, perejil y menta podrían aprovecharse en promociones secundarias o recomendaciones personalizadas orientadas a nichos más estrechos.

Figura 4.6. Gráfico de coordenas paralelas para las reglas que tiene como antecedente el ítem HERB MARKER ROSEMARY.



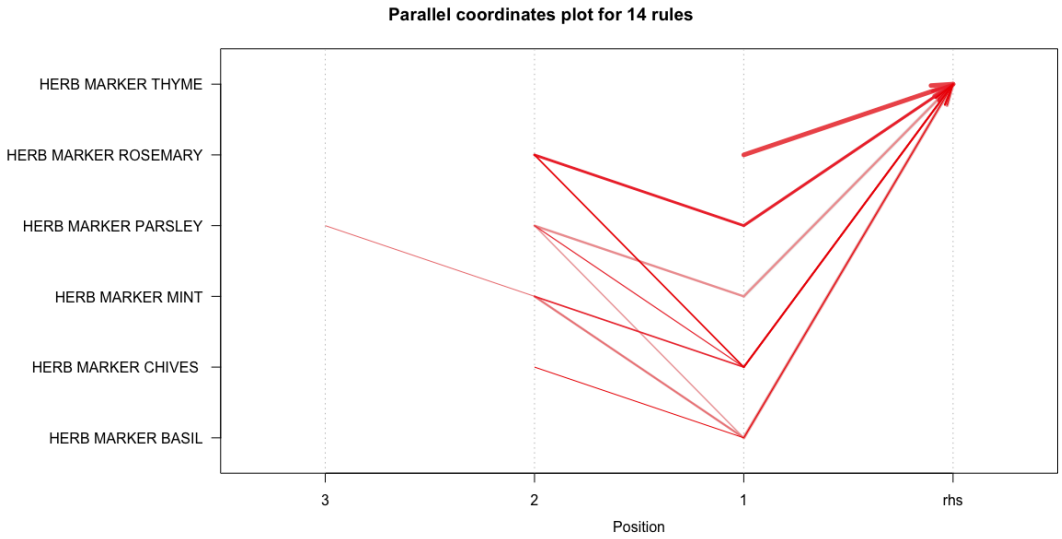
Fuente: elaboración propia.

La Figura 4.6 fue generada con el siguiente código:

```
plot(regla_rosemary_lhs, method = "paracoord")
```

Para terminar de entender cómo se interpretan los gráficos de coordenadas paralelas veamos ahora el ejemplo de las reglas de asociación detectadas con el algoritmo *Apriori* (umbral para el soporte de 0.01 y para la confianza de 0.8) con el ítem "HERB MARKER THYME" como consecuente (**RHS**). Ver Figura 4.7.

Figura 4.7. Gráfico de coordenas paralelas para las reglas que tiene como consecuente el ítem HERB MARKER THYME.



Fuente: elaboración propia.

La Figura 4.7 muestra 14 reglas que confluyen en "HERB MARKER THYME" como consecuente (**RHS**). Las longitudes variables de las flechas evidencian el número de artículos que componen cada antecedente: las más cortas corresponden a reglas de un solo ítem. Por ejemplo, la flecha que sale de "HERB MARKER ROSEMARY" alcanza directamente la columna (**LHS**) Y las flechas más extensas involucran dos o tres marcadores; como por ejemplo, la regla que tiene 3 ítems como antecedente: "HERB MARKER PARSLEY", "HERB MARKER MINT" y "HERB MARKER BASIL"[3]. El grosor y la intensidad del rojo reafirman la jerarquía observada anteriormente: la regla "HERB MARKER CHIVES" $\rightarrow$ "HERB MARKER THYME" sobresale tanto en soporte como en confianza, seguida por la regla con antecedentes "HERB MARKER CHIVES" y "HERB MARKER ROSEMARY". Las flechas tenues, aquellas que parten de "HERB MARKER PARSLEY", "HERB MARKER MINT" o "HERB MARKER BASIL", indican vínculos más débiles; no obstante, su presencia sugiere oportunidades de venta cruzada en segmentos de nicho, especialmente

[3]Esto lo puedes constatar con la linea de código `inspect(regla_thyme_rhs, by = çonfidence")`, es la última regla.

cuando el cliente ya ha mostrado preferencia por marcadores de hierbas múltiples. Así, este gráfico de coordenadas paralelas complementa la lectura estratégica: priorizar *bundles* simples de alto desempeño (la regla "HERB MARKER CHIVES" → "HERB MARKER THYME") y reforzar recomendaciones adicionales cuando el carrito contiene combinaciones menos frecuentes, elevando el ticket promedio sin sacrificar relevancia para el comprador.

La Figura 4.7 fue generada con el siguiente código[4]:

```
plot(regla_thyme_rhs, method = "paracoord")
```

Intenta ahora interpretar el correspondiente a todas las reglas de asociación detectadas con el algoritmo *Apriori* (umbral para el soporte de 0.01 y para la confianza de 0.8) que se obtiene con el siguiente código [5]:

```
plot(head(sort(reglas, by = "support"), 20), method = "paracoord")
```

Pese a que los gráficos de coordenadas paralelas son recursos efectivos para visualizar reglas de asociación, su comprensión demanda un cierto grado de conocimiento en técnicas cuantitativas y experiencia en el estudio de datos. Por lo tanto, si el público objetivo no ha desarrollado completamente estas habilidades, podría ser aconsejable elegir visualizaciones más intuitivas y fácilmente accesibles.

Una visualización de las reglas que puede ser más intuitiva son los grafos[6]. Podemos emplear el motor del paquete igraph (Csardi y Nepusz, 2006) para generar un grafo (**method = "graph"**). En este caso, los vértices se etiquetan con los nombres de los ítems, y los **itemsets** o reglas se representan como un segundo conjunto de vértices. Los elementos se conectan con los **itemsets** o reglas mediante flechas dirigidas. Las flechas que apuntan de los elementos a los vértices de las reglas indican los elementos del **LHS** y una flecha de una regla a un elemento indica el **RHS**. El tamaño y el color de los vértices representan medidas de interés como el *lift* y el soporte. La siguiente linea de código genera la Figura 4.8:

```
plot(regla_thyme_rhs, method = "graph", engine = "igraph")
```

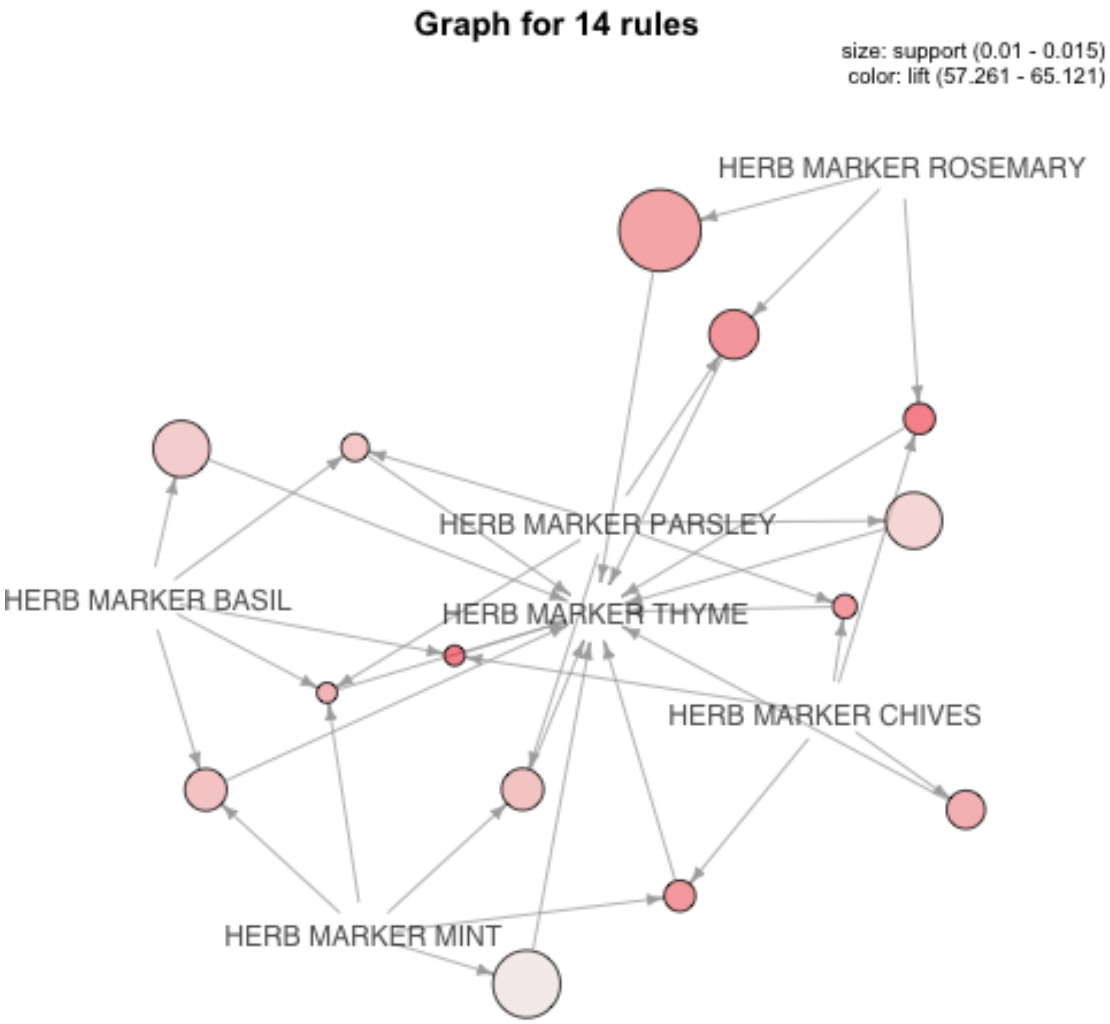
La Figura 4.8 representa en un grafo dirigido las 14 reglas que tienen como consecuente (**RHS**) "HERB MARKER THYME". Como se mencionó arriba, cada círculo corresponde a un ítem y su tamaño refleja el soporte, mientras que la intensidad del rojo codifica el *lift*, que en este caso va de 57 a 65. El resultado es una red donde "HERB MARKER ROSEMARY" destaca por su diámetro al ser el antecedente con mayor frecuencia (soporte) y por un tono rojizo intermedio que denota un *lift* grande pero no extremo. Las flechas grises orientadas hacia "HERB MARKER THYME" hacen visible la direccionalidad **LHS** → **RHS**. Las aristas (líneas que unen los círculos) más cortos provienen de antecedentes simples (por ejemplo, "HERB MARKER ROSEMARY" o "HERB MARKER

[4]Por razones de espacio no se reporta esta visualización.

[5]Si visualizas todas las reglas, tu computador se demorará mucho y en algunos casos colapsará. Por eso filtramos las reglas más importantes según el soporte.

[6]Si no estás familiarizado con los grafos, puedes encontrar una introducción al tema en Alonso y Carabali (2019).

Figura 4.8. Gráfo para las reglas que tiene como consecuente el ítem HERB MARKER THYME.



Fuente: elaboración propia.

CHIVES"), mientras que los ramales que convergen desde "HERB MARKER PARSLEY", "HERB MARKER MINT" y "HERB MARKER BASIL" confirman reglas con múltiples ítems en el antecedente. Esta es otra forma de visualizar lo que ya habíamos visto en el gráfico de coordenadas paralelas (Ver Figura 4.7). No importa cómo lo visualicemos, los *insights* deben ser los mismos[7].

En estos casos, quizás será mejor emplear una visualización interactiva que nos permita "jugar" con el grafo. Para lograr esto, solo tenemos que añadir un "motor" que nos permita generar grafos interactivos. Por ejemplo, podemos emplear **engine = "htmlwidget"**. Con esta aproximación podríamos visualizar las 14 reglas que tienen como consecuente el ítem HERB MARKER THYME; el siguiente código generó la Figura 4.9:

```
# grafo interactivo motor htmlwidget
plot(regla_thyme_rhs, method = "graph", engine = "htmlwidget")
```

En esta versión pdf del libro la Figura 4.9 no será interactiva. Si deseas ver la versión interactiva, puedes consultar la versión html del libro (http://www.icesi.edu.co/editorial/intro-analisis-canastas-web/).

Esta visualización solo será interactiva en la versión web de este libro. El menú desplegable de la esquina superior izquierda de la Figura 4.9 permite seleccionar una regla o un elemento. Al seleccionar, por ejemplo, "HERB MARKER BASIL", el ítem "HERB MARKER BASIL" y todas las reglas asociadas se resaltan en el grafo. Del mismo modo, al seleccionar la `regla 2`, se resalta esta regla y todos los elementos asociados. Esta forma de filtrar elementos y reglas puede ser útil, especialmente cuando el número de reglas es muy grande. ¡Juega un rato con esta visualización!

Pero hay que tener cuidado con los grafos porque tienden a congestionarse a medida que aumenta el número de reglas. Por lo tanto, es mejor visualizar un número relativamente pequeño de reglas con los grafos.

Por ejemplo, graficar en un grafo las 294 reglas de asociación detectadas con el algoritmo *Apriori* (umbral para el soporte de 0.01 y para la confianza de 0.8) sería poco útil. Pero podemos ordenar las reglas según una métrica específica y visualizarlas con las mayores métricas. Por ejemplo, intenta el siguiente código para visualizar 30 reglas[8]:

```
plot(head(sort(reglas, by = "support"), 30), method = "graph", engine =
↪  "htmlwidget")
```

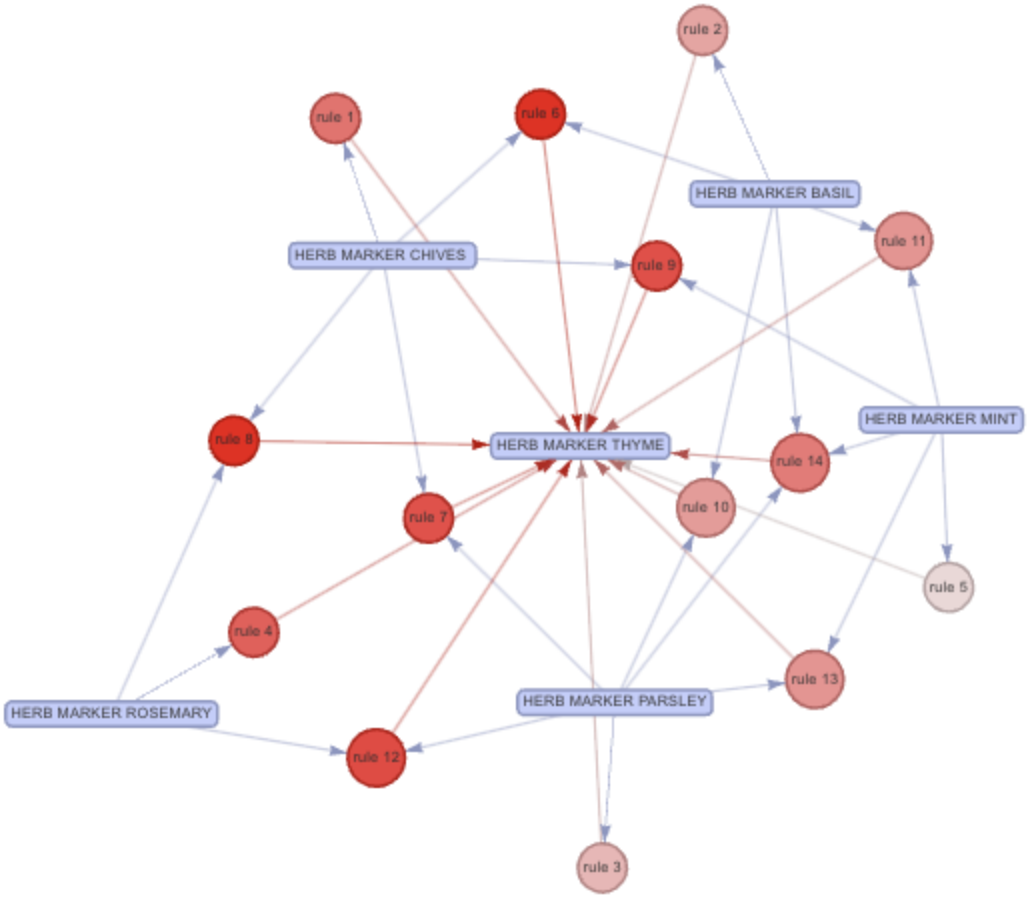
Ahora intenta con 60 reglas:

```
plot(head(sort(reglas, by = "support"), 60), method = "graph", engine =
↪  "htmlwidget")
```

[7] Por un lado, enfatizar la coexhibición "HERB MARKER ROSEMARY" y "HERB MARKER THYME" en el punto de venta por su alta frecuencia y *lift* robusto. Y por otro lado, usar recomendaciones orientadas a nichos para "HERB MARKER MINT", "HERB MARKER BASIL" o "HERB MARKER PARSLEY" cuando el cliente ya lleva varios marcadores en su carrito, optimizando el *cross-selling* sin saturar al comprador

[8] Por razones de espacio no se reporta esta visualización.

Figura 4.9. Grafo interactivo para las reglas que tiene como consecuente el ítem HERB MARKER THYME.



Fuente: Cálculos propios empleando.

Podemos guardar el grafo interactivo empleando la función **saveWidget()** del paquete *htmlwidgets* (Vaidyanathan et al., 2022). En este caso solo tenemos que guardar el grafo en un objeto y emplear la función para guardarlo en un archivo de formato `.html`. Por ejemplo:

```
# Cargar el paquete
library(htmlwidgets)
# guardar el grafo en un objeto
grafo <- plot(head(sort(reglas, by = "support"), 60), method = "graph", engine
↪  = "htmlwidget")

# Grabando el archivo
saveWidget(grafo, file = "reglas_60_grafo.html")
```

¡Esta visualización interactiva de los resultados en forma de grafo puede ser muy útil para la toma de decisiones! No obstante, sigue requiriendo cierto grado de sofisticación cuantitativa del tomador de decisiones. En algunas ocasiones, será más conveniente entregar toda la información de las reglas de asociación encontradas para que el tomador de decisiones tenga toda la información relevante. La función inspectDT() del paquete arulesViz (Hahsler, 2017) genera un widget HTML que permite jugar de forma interactiva con el conjunto de reglas de asociación que tengamos guardadas en un objeto, presentándolo en forma de cuadro interactivo.

Por ejemplo, el siguiente código genera una tabla en formato `.html` que permite interactuar con las reglas.

```
# Tabla con resultados
inspectDT(reglas)
```

En esta versión pdf del libro la Figura 4.10 no será interactiva. Si deseas ver la versión interactiva, puedes consultar la versión html del libro (http://www.icesi.edu.co/editorial/intro-analisis-canastas-web/). Este tipo de tabla interactiva permite al usuario interactuar con los datos. Nota que esta tabla también la puedes grabar como un archivo `.html` empleando la función **saveWidget()** que vimos anteriormente.

Otra opción para entregar los resultados al usuario, pero un poco más complicada de distribuir[9], es emplear el paquete *shiny*[10] (Chang et al., 2021) y la función **ruleExplorer()**. ¡Intenta el siguiente código y juega con el resultado!

```
# Cargar el servidor shiny
ruleExplorer(reglas)
# Para ver el servidor, pega la dirección correspondiente en un navegador
↪  busca
# la dirección des pues del texto 'Listening on' copia lo que sigue debe
# iniciar por 'http://'
```

[9]En este caso tendremos que instalar un servidor **shiny**, pues se necesita que R esté corriendo para que la *app* de *shiny* esté corriendo.

[10]Este es un paquete de R que permite generar rápidamente aplicaciones web.

Figura 4.10. Widget con reglas de asociación generadas con el algoritmo Apriori.

Show 10 entries Search:

	LHS	RHS	support	confidence	cover
	All	All	All	All	All
[1]	{HERB MARKER CHIVES }	{HERB MARKER BASIL}	0.0100608948901244	0.844444444444444	0.0119142176330
[2]	{HERB MARKER CHIVES }	{HERB MARKER PARSLEY}	0.0105904156738152	0.888888888888889	0.0119142176330
[3]	{HERB MARKER CHIVES }	{HERB MARKER ROSEMARY}	0.0108551760656606	0.911111111111111	0.0119142176330
[4]	{HERB MARKER CHIVES }	{HERB MARKER THYME}	0.0113846968493513	0.955555555555556	0.0119142176330
[5]	{HERB MARKER CHIVES }	{HERB MARKER MINT}	0.011119936457506	0.933333333333333	0.0119142176330
[6]	{CHILDS GARDEN FORK BLUE }	{CHILDS GARDEN TROWEL BLUE }	0.0105904156738152	0.888888888888889	0.0119142176330
	{CHILDS	{CHILDS			

Fuente: elavoración propia.

Ahora estamos listos para interpretar y diseñar estrategias de mercadeo. ¡La imaginación es el límite!

4.4 Comentarios finales

Este capítulo discute la visualización de los resultados del **MBA**, el insumo principal para entregar a quienes toman las decisiones en la organización. Así, la tarea de visualización consiste en usar gráficos que "cuenten" la historia de lo observado en las canastas de compras. Como se ha discutido en capítulos anteriores, aunque son roles diferentes, el científico de datos y el *analytics translator* deben trabajar de manera coordinada.

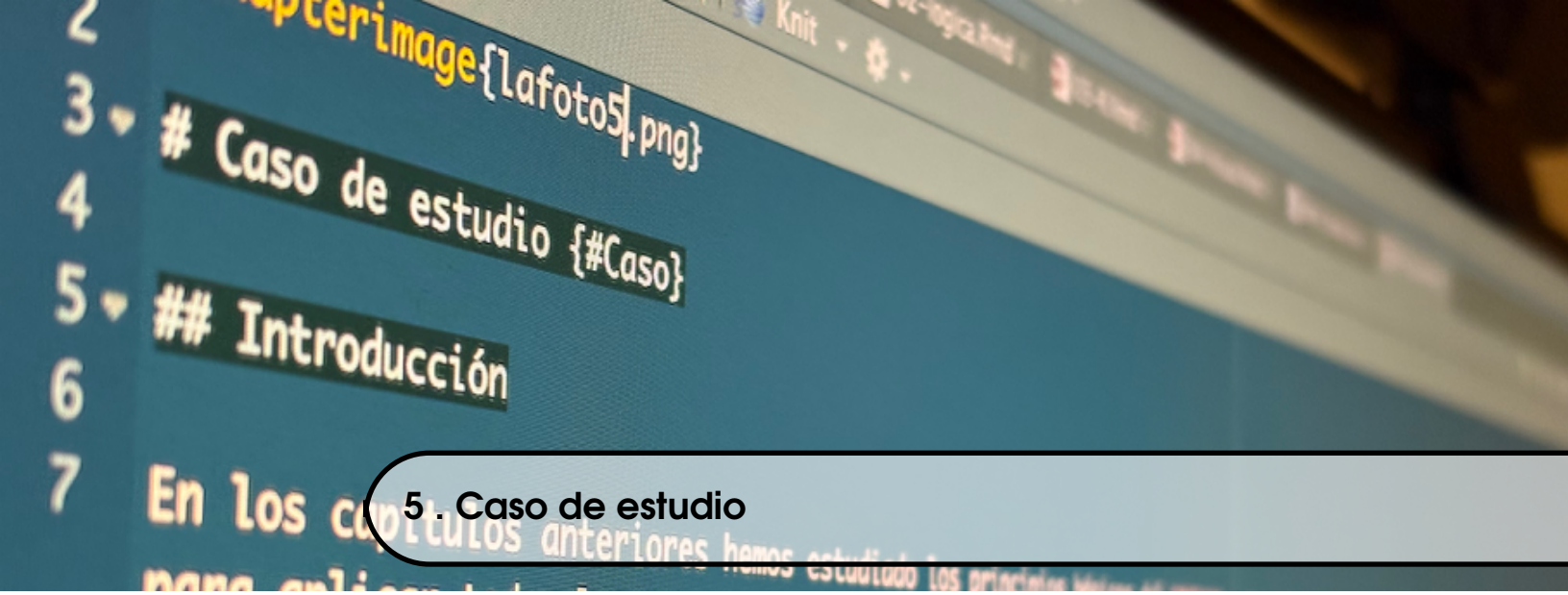
Este capítulo le permitirá al científico de datos saber cómo realizar la codificación para la visualización de los resultados. El científico de datos debe entender las preguntas del negocio para generar visualizaciones coherentes con estas necesidades. En el caso del *analytics translator*, el contenido de este capítulo le permitirá conocer qué tipo de visualización solicitar al científico de datos. El *analytics translator* debe entender las posibilidades que hay para visualizar los resultados de tal forma que facilite la toma de decisiones.

El **MBA** es una herramienta del *business analytics* que permite tomar decisiones a partir de datos que surgen de las transacciones habituales en los consumidores de un canal minorista. Teniendo en cuenta las decisiones de negocio que se esperan tomar, el **MBA** crea reglas de asociación que facilitan tomar decisiones basados en datos. Es decir, el **MBA** supone que el minorista ha consolidado una base de datos con las transacciones de los clientes y que tiene una pregunta estratégica de negocio que permite enfocar las reglas de asociación. El **MBA** resuelve múltiples preguntas de negocio (ver sección 1.1) que impliquen encontrar la coocurrencia de productos en las canastas de compra.

Las herramientas discutidas en este libro pueden ser utilizadas por científicos de datos y *analytics translators* para pasar de datos transaccionales a la toma de decisiones empleando el **MBA**. Los científicos de datos deben entender las preguntas de negocio para generar el análisis pertinente y las visualizaciones que permitan comunicar los resultados obtenidos. Los *analytics translators* deben entender las potencialidades y limitaciones del **MBA** para poder traducir las preguntas de negocio al equipo de científicos de datos. Así mismo, al culminarse el **MBA** será necesario escoger qué tipos de visualizaciones solicitar a los científicos de datos para facilitar la toma de decisiones. Como hemos visto a lo largo de este libro, el **MBA** es una herramienta poderosa para tomar decisiones basadas en datos en un entorno minorista.

Las aplicaciones del **MBA** son muchas, ¡la imaginación es el límite!

En el siguiente capítulo veremos un caso completo de estudio aplicado a datos de una panadería en línea de Edimburgo. Ese capítulo te podrá dar ideas de cómo aproximarte a tu primera pregunta de negocio que se pueda responder con la tarea de *Encontrar Reglas de Asociación*.



5 . Caso de estudio

5.1 Introducción

En los capítulos anteriores hemos estudiado los principios básicos del **MBA** y cómo implementar en R este tipo de análisis. En este Capítulo presentamos un estudio de caso para aplicar todos los conceptos presentados en los capítulos anteriores y mostrar cómo el proceso de exploración de datos puede hacer replantear o precisar las preguntas de negocio.

Antes de iniciar, es importante recordar que, como lo estudiamos en el Capítulo 1, el proceso del *business analytics* permite pasar de datos a *insights* para la toma de decisiones. Este proceso parte de una pregunta de negocio que es definida por la organización con ayuda del *analytics translator*. Una vez que la pregunta está definida y se cuenta con los datos, las primeras actividades son la limpieza y preparación de los datos, su exploración y visualización y el modelado que se requiera para responder a la pregunta de negocio planteada. Estas actividades son responsabilidad del científico de datos. Posteriormente, los *insights* descubiertos en el modelado deben ser comunicados a los tomadores de decisiones; esta actividad es responsabilidad principal del *analytics translator*.

Este Capítulo está organizado siguiendo el orden de estas actividades. Primero se presentarán los resultados de cada una de las fases, de tal manera que aquellos de ustedes que se encuentren (o estén interesados) en el rol de *analytics translator* se concentren en los resultados y los *insights*. En el Anexo de este Capítulo, los interesados en el rol de científico de datos podrán encontrar todo el código que se emplea en cada una de las actividades.

5.2 El contexto y la pregunta de negocio

Para este caso emplearemos una base de datos que contiene transacciones para la panadería en línea *The Bread Basket* de Edimburgo (Escocia), que han sido publicadas por Mittal (2018). La panadería *The Bread Basket* se encuentra en el casco histórico

de Edimburgo, y se ha convertido en punto de "peregrinación" para los amantes del pan artesanal (Hery y Widjaja, 2024). Su propuesta fusiona técnicas clásicas británicas con sabores de Argentina y España (Oliveira, 2018). En sus vitrinas conviven *baguettes* recién horneadas con alfajores de dulce de leche y ensaimadas mallorquinas. La oferta de productos argentinos y españoles de esta panadería está descrita en varios estudios que emplean datos (Egbeola, 2023; Oliveira, 2018).

Entre enero de 2016 y diciembre de 2017, la panadería registró 9465 transacciones y se vendieron 94 ítems diferentes. Para ese entonces, *The Bread Basket* funcionaba principalmente durante el día (de media mañana a primeras horas de la tarde) y combinaba la atención tradicional en tienda con un servicio de pedidos en línea. La panadería registró meticulosamente cada venta, y la hora y día de esta. Algunos autores como Egbeola (2023) indican que los clientes podían ordenar sus productos a través de la web y recogerlos en la tienda, lo cual era un elemento innovador para un negocio de este tipo en esa época.

Además de panes y repostería, la tienda ofrecía experiencias especiales y productos promocionales. Por ejemplo, organizaba una sesión titulada "*Afternoon with the Baker*" (Tarde con el panadero) para interactuar con clientes, y vendía artículos de *merchandising* como camisetas y tarjetas de ocasión (por ejemplo, tarjetas de San Valentín)[1]. El detalle de cada transacción revela que la panadería presentaba un modelo híbrido que combina cafetería de barrio, tienda gourmet y microeventos gastronómicos.

La panadería, como la mayoría de negocios, estaba en ese momento buscando incrementar sus ventas. Para lograr su objetivo, el gerente tenía las siguientes preguntas de negocio[2]:

- ¿Tiene sentido hacer un descuento en el café?
- ¿Cuál sería un combo que dispare las ventas?
- ¿Qué recomendaciones se pueden hacer para vender más té?

El archivo `bread basket.csv` contiene cada una de las transacciones de esta panadería en línea entre el 30 de octubre del 2016 y el 9 de abril de 2017. Los datos originalmente fueron descargados de https://www.kaggle.com/datasets/mittalvasu95/the-bread-basket?resource=download (Mittal, 2018). Los datos que se emplean a continuación están disponibles en la página web del libro en el archivo `bread basket.csv`.

La base de datos contiene 20507 registros correspondientes a 9465 transacciones realizadas. La base de datos cuenta con 5 columnas, que representan igual número de variables. Las variables son:

- `Transaction`: identificador único y numérico de la transacción. Esta variable permite agrupar los distintos ítems que componen una misma compra.
- `Item`: Nombre del producto adquirido.

[1] Esta conclusión surge de la misma base de datos que estudiaremos más adelante.

[2] Estas preguntas de negocio son ficticias. Hasta aquí la narración corresponde a hechos reales documentados por otros autores y soportados en la base de datos. Las preguntas de negocio son construidas para efectos pedagógicos de este capítulo y no existe ninguna evidencia sobre cuál era la real pregunta de negocio o si existía alguna.

- `date_time`: Marca de fecha y hora exactas del registro (formato dd-mm-aaaa hh:mm).
- `period_day`: Franja del día en la que ocurre la venta (*Morning, Afternoon*, *Evening* o *Night*).

- `weekday_weekend`: Variable que marca si la transacción se realizó en días laborales (*Weekday*) o fines de semana (*Weekend*).

5.3 Exploración de los datos

Tras cargar los datos[3], el científico de datos empezó a explorarlos. Una de las primeras ideas que se le ocurrió, fue crear un cuadro que permitiera observar el porcentaje de las transacciones que se hacen por tipo de día y momento del día. Los resultados se reportan en el Cuadro 5.1.

Tabla 5.1. Distribución de las transacciones por tipo de día y período del día (porcentaje del total de transacciones)

	weekday	weekend	Sum
afternoon	35.13	18.64	53.77
evening	1.79	0.97	2.76
morning	27.98	15.37	43.35
night	0.03	0.10	0.13
Sum	64.92	35.08	100.00

Fuente: elaboración propia.

Solo el 0.13 % de las transacciones se realizan en la medianoche[4] (Ver Cuadro 5.1 fila *night*). Y las transacciones de la noche (*evening*) representan el 2.76 % de las transacciones totales (Ver Cuadro 5.1 fila *evening*). Es decir, entre las 6 p.m. y las 6 a.m. (*evenning* y *night*) solo se presentan el 2.89 % de las transacciones.

Estos resultados del Cuadro 5.1 permitieron al científico de datos empezar a "sospechar" que se deberían analizar los datos por periodo del día y no todos en conjunto. Los resultados mostraban que la mayoría de las transacciones ocurren en la tarde (*afternoon*) y las mañanas (*morning*). Por lo tanto, debería concentrar la atención en esos periodos. Tras un chequeo rápido con el *analytics translator* acordaron que tenía sentido seguir explorando esa idea de concentrar la atención en esos periodos. El conocimiento del negocio del *analytics translator* le permitía entender que el comportamiento de los compradores y sus gustos eran muy diferentes de noche y no valía el esfuerzo analizar ese periodo. Por eso decidieron concentrar la atención en lo que

[3] La narración de esta, y de las siguientes secciones, es ficticias y fue construida recreando lo que los autores consideramos una interacción común entre un *analytics translator* y un equipo de científicos de datos.

[4] Es importante recordar que típicamente el horario de Evening va de 6 p.m. a 12 a.m y el del periodo *night* va de 12 a.m. a 6 a.m.

ocurre en la tarde (*afternoon*) y las mañanas (*morning*)[5].

Después de esa decisión surgieron varias preguntas nuevas que inquietaban al *analytics translator* para poder comunicar su decisión a los tomadores de decisiones:

- ¿Cuántas observaciones tendremos ahora?
- ¿Cuántos productos se pierden con esta decisión?

El científico de datos encontró rápidamente la respuesta a estas preguntas. Hay 4 ítems que se venden en los periodos de *night* y *evening* que no se venden en el resto del día. Y pasamos de 9,465 a 9,192 transacciones, es decir, 273 transacciones menos. Esto no es sustancial y permite que nuestro análisis se concentre en dos periodos del día.

Entonces procedió a actualizar el Cuadro 5.1, sin las transacciones de los periodos de *night* y *evening*, obteniendo el Cuadro 5.2.

Tabla 5.2. Distribución de las transacciones por tipo de día y período del día para la base filtrada (porcentaje del total de transacciones)

	weekday	weekend	Sum
afternoon	36.17	19.19	55.36
morning	28.81	15.83	44.64
Sum	64.98	35.02	100.00

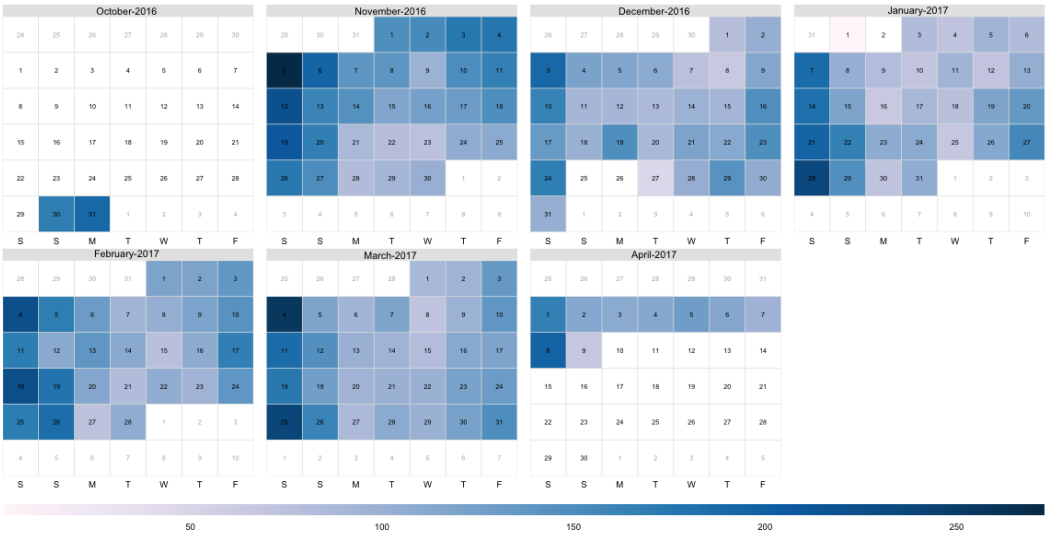
Fuente: elaboración propia.

La siguiente exploración que realizó el científico de datos tenía que ver con los días de semana y los de los fines de semana. Para eso empezó a crear un gráfico de calendario (Calendar plot)[6] (Ver Figura 5.1).

[5]Los horarios de la mañana (*morning*) y tarde (*afternoon*) típicamente van de 6 a.m. a 12 p.m. y de 12 p.m. a 6 p.m., respectivamente.

[6]El gráfico de calendario es una visualización de datos de series de tiempo que utiliza un formato de calendario para mostrar información temporal (Alonso y Largo, 2023). A diferencia de los gráficos de líneas, el gráfico de calendario muestra los datos en un contexto temporal natural, facilitando la identificación de patrones como estacionalidad o ciclos. Así mismo, esta visualización permite comparar fácilmente diferentes períodos (meses, años) al colocarlos uno al lado del otro. Adicionalmente, al emplear el formato familiar del calendario, esta visualización facilita la comprensión de los datos a personas sin experiencia en análisis de datos (Alonso y Largo, 2023).

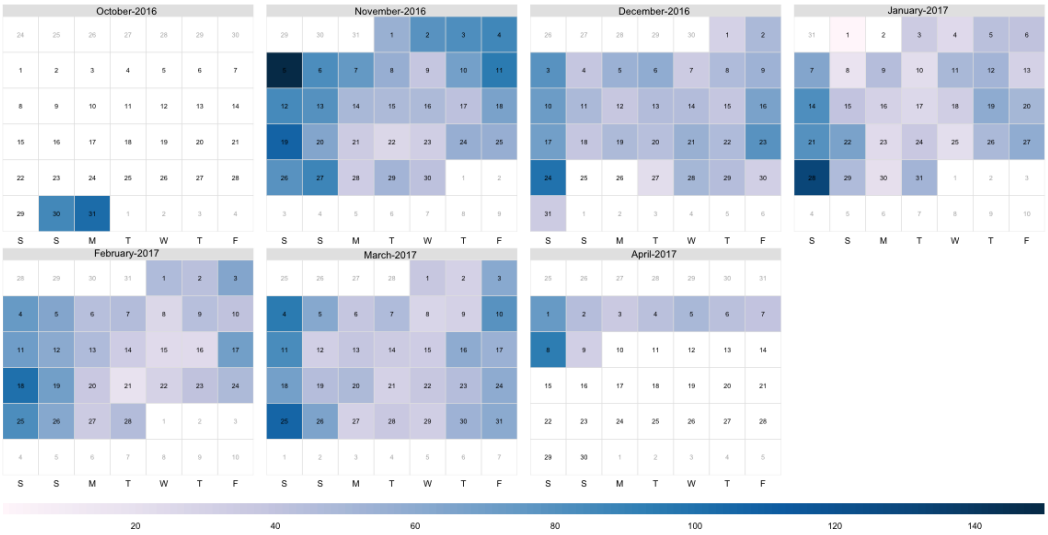
Figura 5.1. Número de transacciones de la mañana y de la tarde de la panadería



Fuente: elaboración propia.

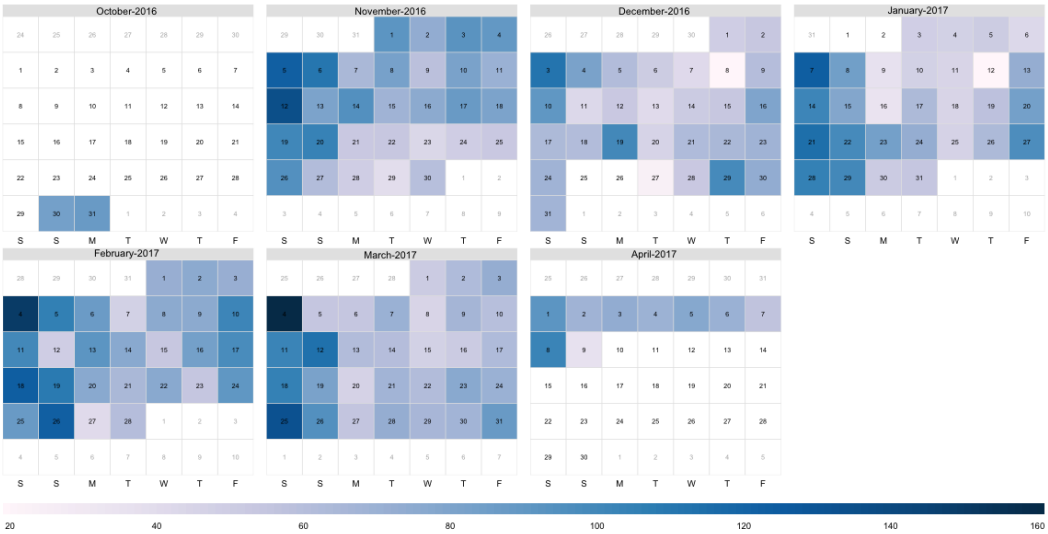
Así mismo, el científico de datos construyó las Figuras 5.2 y 5.3, que muestran las transacciones solo para la mañana y solo para la tarde, respectivamente.

Figura 5.2. Número de transacciones de la mañana de la panadería



Fuente: elaboración propia.

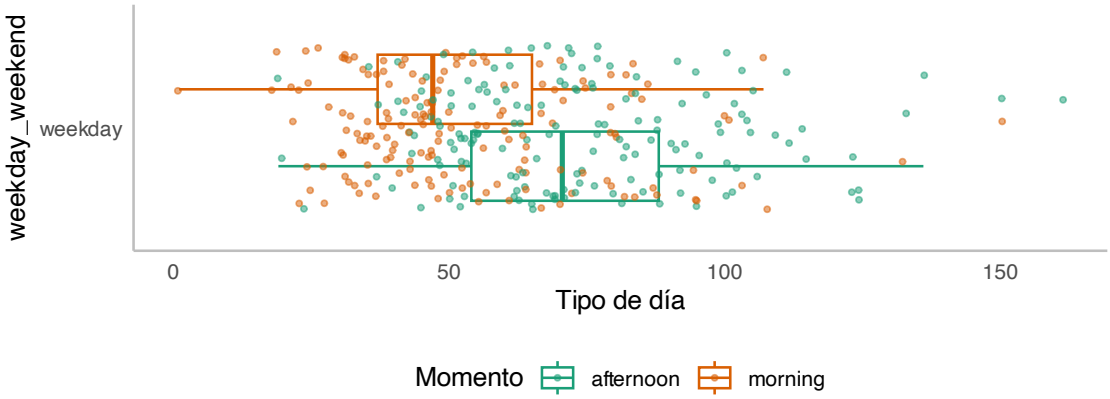
Figura 5.3. Número de transacciones de la tarde de la panadería



Fuente: elaboración propia.

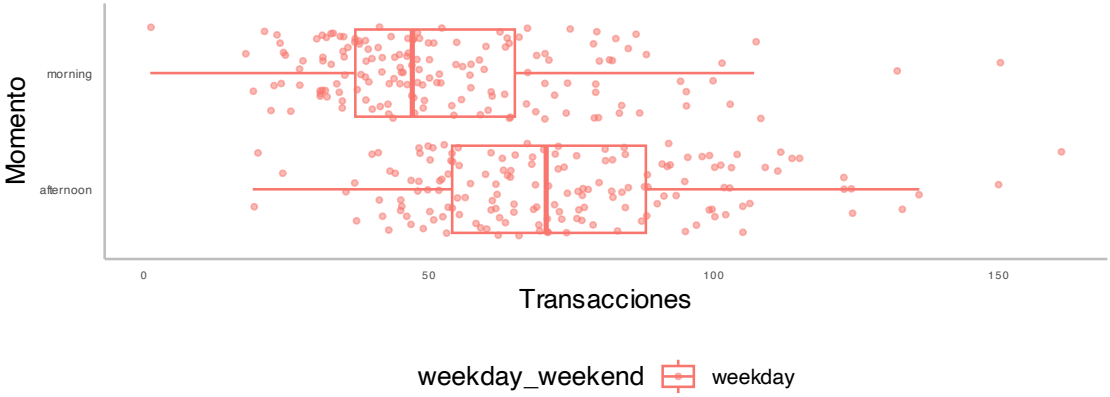
De estas visualizaciones, el científico de datos intuyó que existía una diferencia en el comportamiento del número de transacciones entre los fines de semana y durante los días laborales. Otra manera de ver esa diferencia fue empleando un *boxplot*, como en la Figura 5.4 y la Figura 5.5.

Figura 5.4. Distribución del número de transacciones de la panadería por tipo de día y momento del día



Fuente: elaboración propia.

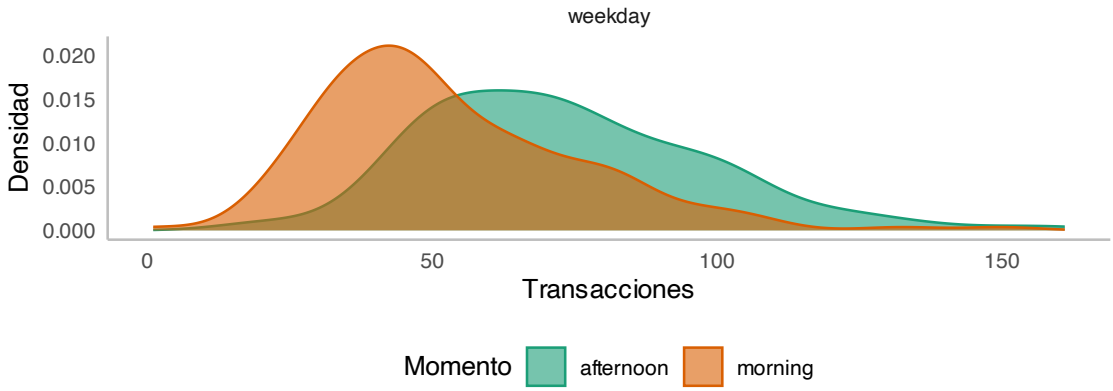
Figura 5.5. Distribución del número de transacciones de la panadería por momento del día y tipo de día



Fuente: elaboración propia.

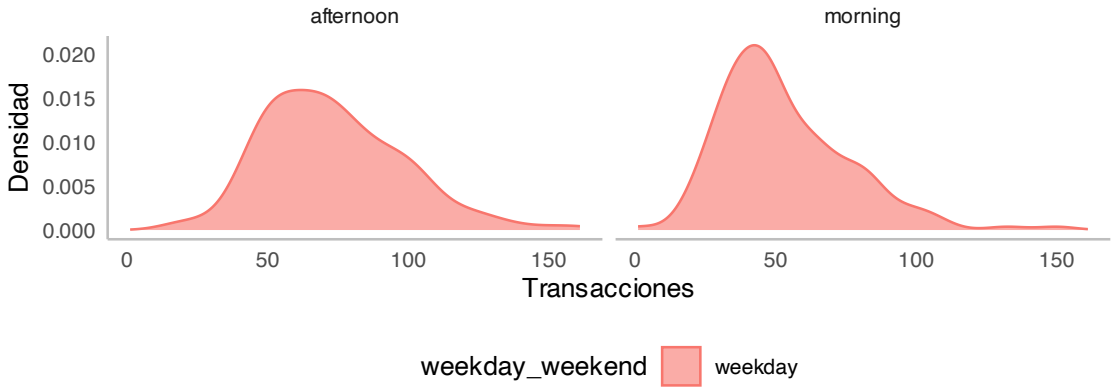
Otras visualizaciones alternas se presentan en las Figuras 5.6 y 5.7.

Figura 5.6. Densidad estimada para el número de transacciones de la panadería por tipo de día y momento del día



Fuente: elaboración propia.

Figura 5.7. Densidad estimada para el número de transacciones de la panadería por momento del día y tipo de día

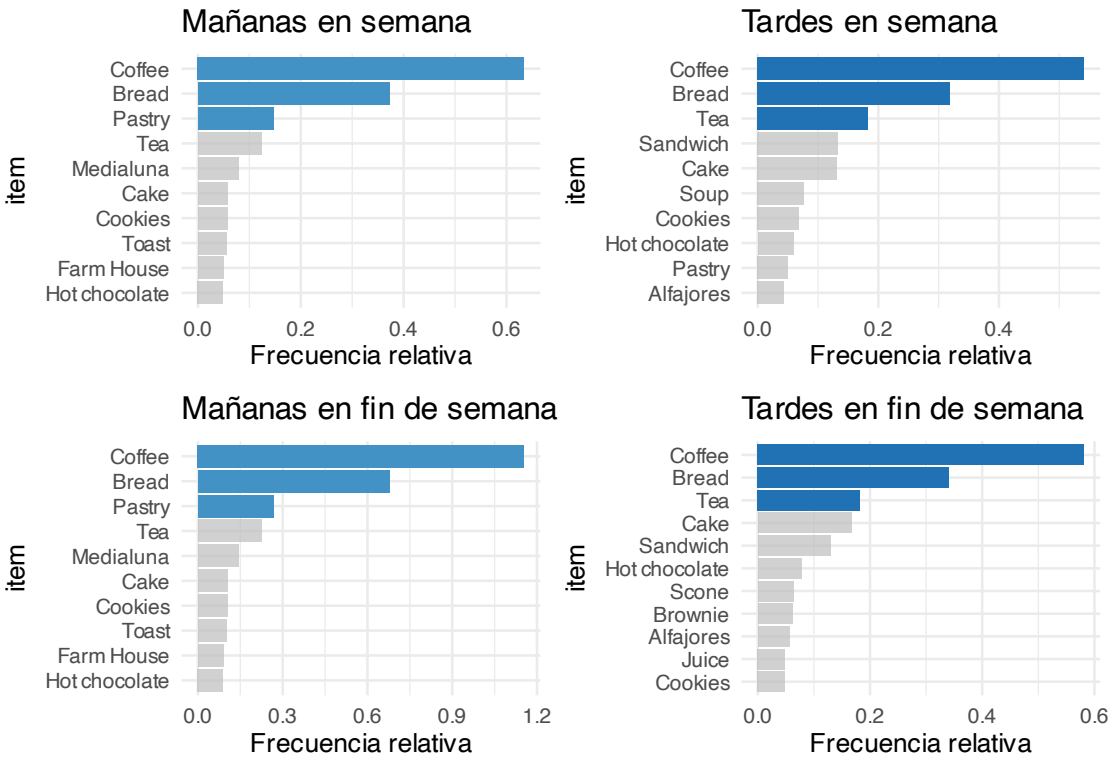


Fuente: elaboración propia.

Las visualizaciones le permitieron intuir al científico de datos que el número de transacciones es diferente por tipo de día y momento del día. Con estos nuevos *insights*, el científico de datos exploró si los productos presentes en las canastas también diferían por tipo de día y momento del día.

Así, el científico de datos construyó visualizaciones para los ítems más frecuentes por momento del día y por tipo de día, como se ve en la Figura 5.8.

Figura 5.8. Ítems más frecuentes en las transacciones de la panadería por momento del día y tipo de día (proporción 0-1)



Fuente: elaboración propia.

El científico de datos notó que los dos primeros ítems más frecuentes son el café y el pan, sin importar la hora o el tipo de día. El tercer ítem es idéntico en las mañanas y en las tardes sin importar el tipo de día. A partir del cuarto ítem ya se empiezan a diferenciar los productos más frecuentes.

5.4 Reformulación de la pregunta de negocio

Claramente, esta exploración mostraba que existe diferencia entre el comportamiento de las ventas, entre el tipo de día (`weekday_weekend`) y el momento del día (`period_day`) y, por tanto, deberían tomarse decisiones diferentes para los días y los momentos del día.

Estos hallazgos llevaron al científico de datos y al *analytics translator* a replantearse las preguntas de negocio iniciales. Tras consultas con los tomadores de decisiones, por

parte del *analytics translator*, se replantearon las preguntas de negocio de tal manera que las nuevas preguntas de negocio fueron:

- ¿Existe alguna diferencia entre las reglas de asociación, dependiendo del tipo de día (`weekday_weekend`) y el momento del día (`period_day`)?
- ¿Tiene sentido hacer un descuento en el café?
- ¿Se puede crear un combo con sentido que sirva cualquier día y en cualquier momento del día?
- ¿Qué recomendaciones se pueden hacer para vender más té?

5.5 Modelado

Con las preguntas ya claras y el conocimiento de los datos y del negocio, el científico de datos procedió a encontrar las reglas por tipo de días y momento del día. El científico de datos, de común acuerdo con el *analytics translator*, decidió emplear umbrales para el soporte de 1% (0.01) y para la confianza de 25% (0.25). Así mismo, decidieron solo reportar reglas con, por lo menos, un ítem en el **RHS** y en el **LHS**.

Se encontraron el siguiente número de reglas no redundantes:

- 21 para los días de la semana en la tarde,
- 37 para los fines de semana en la tarde,
- 17 para los días de la semana en la mañana y
- 26 para los fines de semana en la mañana.

Con las reglas encontradas para cada uno de estos grupos, el científico de datos y el *analytics translator* procedieron a analizar los resultados.

5.6 Resultados por tipo y momento del día

Para poder encontrar los *insights* para el negocio, el científico de datos preparó tablas y visualizaciones que permitieran entender mejor las reglas encontradas. Con estos insumos, se sentaron con el *analytics translator* a analizar los resultados.

5.6.1 Reglas para los días de la semana por la tarde

Las reglas identificadas por el científico de datos para los días de la semana por la tarde se reportan en el Cuadro 5.3. En las Figuras 5.9 y 5.10 se presentan visualizaciones de las métricas de las reglas.

Tabla 5.3. Reglas encontradas con el algoritmo Apriori a las transacciones de los días de semana en la tarde según lift

LHS		RHS	Soporte	Confianza	Cobertura	Lift
Soup	=>	Tea	0.02	0.27	0.07	1.62
Toast	=>	Coffee	0.02	0.70	0.02	1.53
Brownie	=>	Tea	0.01	0.25	0.04	1.49

Tabla 5.3. Reglas encontradas con el algoritmo Apriori a las transacciones de los días de semana en la tarde según lift (cont.)

LHS		RHS	Soporte	Confianza	Cobertura	Lift
Salad	=>	Coffee	0.01	0.65	0.02	1.42
Pastry	=>	Coffee	0.03	0.53	0.05	1.16
Cake	=>	Coffee	0.07	0.53	0.13	1.15
Alfajores	=>	Coffee	0.02	0.52	0.04	1.13
Sandwich	=>	Coffee	0.06	0.51	0.12	1.12
Scone	=>	Coffee	0.02	0.51	0.03	1.11
Hot chocolate	=>	Coffee	0.03	0.51	0.06	1.11
Pastry	=>	Bread	0.02	0.33	0.05	1.10
Medialuna	=>	Coffee	0.02	0.50	0.03	1.08
Muffin	=>	Coffee	0.02	0.48	0.04	1.06
Juice	=>	Coffee	0.02	0.47	0.04	1.04
Cookies	=>	Coffee	0.03	0.45	0.07	0.97
Brownie	=>	Coffee	0.02	0.44	0.04	0.96
Alfajores	=>	Bread	0.01	0.27	0.04	0.92
Soup	=>	Coffee	0.03	0.42	0.07	0.91
Chicken Stew	=>	Coffee	0.01	0.39	0.03	0.84
Tea	=>	Coffee	0.05	0.31	0.17	0.68
Bread	=>	Coffee	0.08	0.28	0.30	0.61

Fuente: elaboración propia.

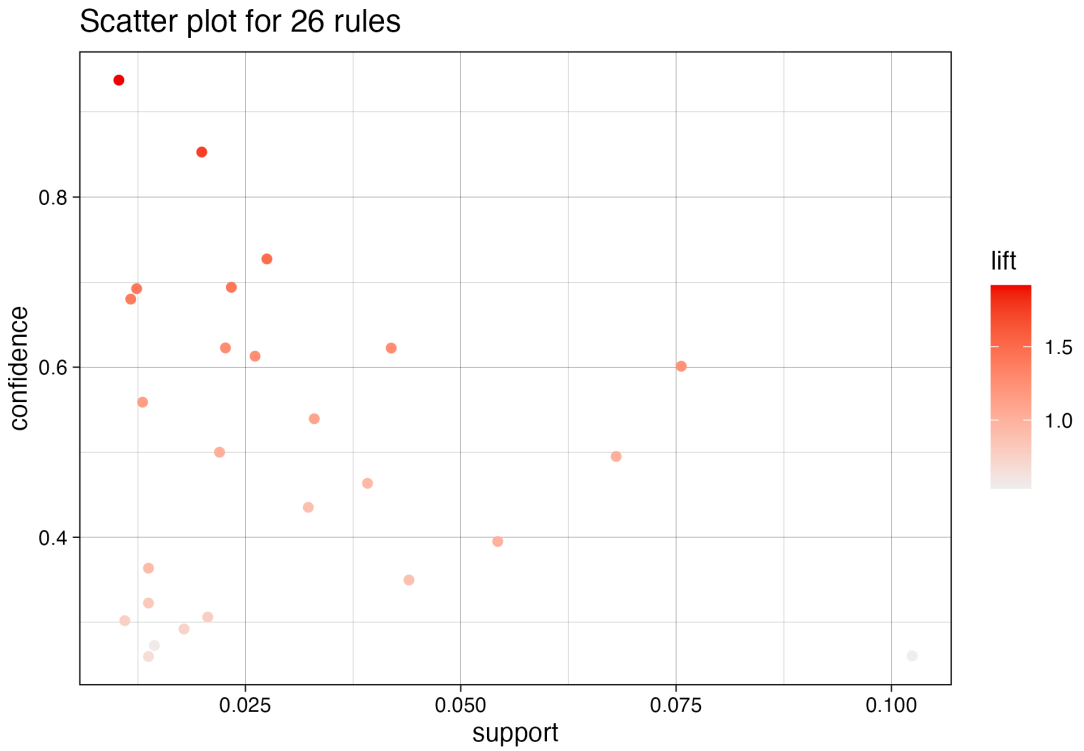
En esta versión en formato PDF del libro, El Cuadro 5.3 no es interactivo. Si deseas ver la versión interactiva, que además contiene todas las reglas, puedes consultar la versión en formato HTML del libro (http://www.icesi.edu.co/editorial/intro-analisis-canastas-web/).

La canasta de las transacciones registradas en la franja de la tarde de los días laborales muestra un patrón de consumo claro: las bebidas calientes, en particular *Coffee*, funcionan como "eje" de la mayoría de las 21 reglas detectadas. Según el Cuadro 5.3„ ocho de las diez reglas con mayor *lift* presentan a *Coffee* como consecuente directo y las dos restantes a *Tea*, lo que confirma que los productos de pastelería suelen comprarse como acompañantes de la bebida estrella. Un ejemplo es la regla $\{Pastry\} \rightarrow \{\text{Coffee}\}$, cuyo *lift* es 1.41, la confianza 0.64 y la cobertura 0.14; es decir, la canasta ocurre un 41 % más de lo esperado si los ítems fueran independientes y aparece en el 14 % de las transacciones de la tarde de los días laborales.

Aunque el soporte individual de los artículos rara vez supera el 3 %, el *lift* elevado indica que el emparejamiento no es cuestión del azar. Este resultado presenta oportunidades para dinamizar ventas mediante promociones dirigidas. En el extremo opuesto, la regla $\{Soup\} \rightarrow \{Tea\}$, con un *lift* de 1.62 y soporte de 0.02, revela un nicho reducido pero coherente: clientes que prefieren una combinación ligera a media tarde. Tal

segmento es ideal para ensayar un "combo" de sopa + té sin canibalizar la demanda principal de café.

Figura 5.9. Visualización de las métricas de las reglas obtenidas al aplicar el algoritmo Apriori a las transacciones de los días de semana en la tarde.

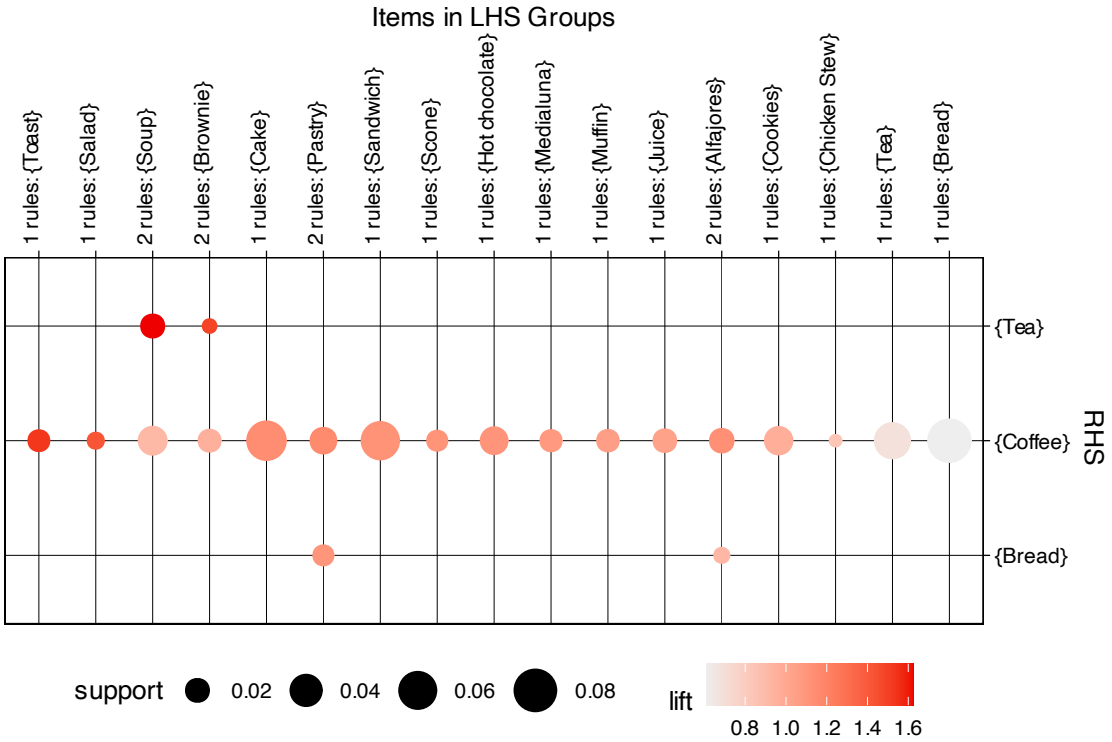


Fuente: elaboración propia.

En este formato PDF del libro la Figura 5.9 no es interactiva. Si deseas ver la versión interactiva, puedes consultar la versión en formato HTML del libro (http://www.icesi.edu.co/editorial/intro-analisis-canastas-web/).

La nube de soporte y *lift* que aparece en la Figura 5.9 muestra dos grupos: reglas con bajo soporte, pero alto *lift* (nichos diferenciadores) y reglas con soporte y confianza medias (combinaciones frecuentes, menos probables). La Figura 5.10 completa el panorama al mostrar al *Coffee* como el consecuente de la mayoría de las reglas encontradas.

Figura 5.10. Visualización alternativa de las reglas obtenidas al aplicar el algoritmo Apriori a las transacciones de los días de semana en la tarde.



Fuente: elaboración propia.

5.6.2 Reglas para los fines de semana por la tarde

Las reglas identificadas por el científico de datos para los fines de semana por la tarde se reportan en el Cuadro 5.4. En las Figuras 5.11 y 5.12 se presentan visualizaciones de las métricas de las reglas.

El análisis de las reglas de asociación para las tardes de los fines de semana encontró 37 reglas. En este caso, las reglas con mayor *lift* están principalmente dirigidas hacia productos de pastelería (*Cake*) y bebidas calientes como el té (*Tea*). La regla con el *lift* más alto es $\{\text{Coffee, Hot chocolate}\} \rightarrow \{Cake\}$, con un valor de 2.07. Este *lift* implica que, cuando se compran conjuntamente café y chocolate caliente, la probabilidad de adquirir pastel se incrementa en un 107 % respecto a lo esperado bajo independencia. De manera similar, otras reglas que apuntan hacia la compra de *Cake* tienen altos valores de *lift*, indicando que en las tardes de fin de semana, el pastel actúa como un complemento habitual de las bebidas calientes.

Por otro lado, aunque con menor *lift* relativo, se destacan las asociaciones cuyo consecuente es *Coffee*, en especial aquellas que provienen de alimentos salados o preparaciones específicas como ensaladas o *brunches* (*Spanish Brunch*, *Salad* y *Hearty & Seasonal*). Estas reglas presentan valores de confianza alrededor del 60 % y valores de *lift* que van desde 1.30 hasta 1.46, indicando asociaciones relevantes pero menos sorprendentes que las dirigidas hacia el pastel.

Además, se identifican reglas que vinculan productos típicos de merienda tradicional, como $\{Scone\} \rightarrow \{Tea\}$, con un *lift* de 1.67 y un soporte cercano al 2 %. Esta regla es particularmente interesante, dado que no apareció en el análisis realizado para días laborales, lo que señala un hábito de consumo exclusivo de fines de semana.

Tabla 5.4. Reglas encontradas con el algoritmo Apriori a las transacciones del fin de semana en la tarde según lift

LHS		RHS	Soporte	Confianza	Cobertura	Lift
Coffee, Hot chocolate	=>	Cake	0.01	0.33	0.03	2.07
Scone	=>	Tea	0.02	0.29	0.06	1.67
Hot chocolate	=>	Cake	0.02	0.26	0.07	1.65
Coffee, Tea	=>	Cake	0.02	0.26	0.07	1.65
Hearty & Seasonal	=>	Coffee	0.01	0.68	0.02	1.46
Spanish Brunch	=>	Coffee	0.02	0.63	0.04	1.35
Salad	=>	Coffee	0.01	0.62	0.02	1.34
Toast	=>	Coffee	0.01	0.62	0.02	1.33
Pastry	=>	Coffee	0.03	0.61	0.04	1.32
Medialuna	=>	Coffee	0.03	0.60	0.04	1.30
Soup	=>	Coffee	0.02	0.60	0.04	1.30
Sandwich	=>	Coffee	0.06	0.59	0.11	1.28
Cake, Hot chocolate	=>	Coffee	0.01	0.59	0.02	1.27
Bread, Hot chocolate	=>	Coffee	0.01	0.58	0.02	1.24
Jammie Dodgers	=>	Bread	0.01	0.39	0.03	1.22

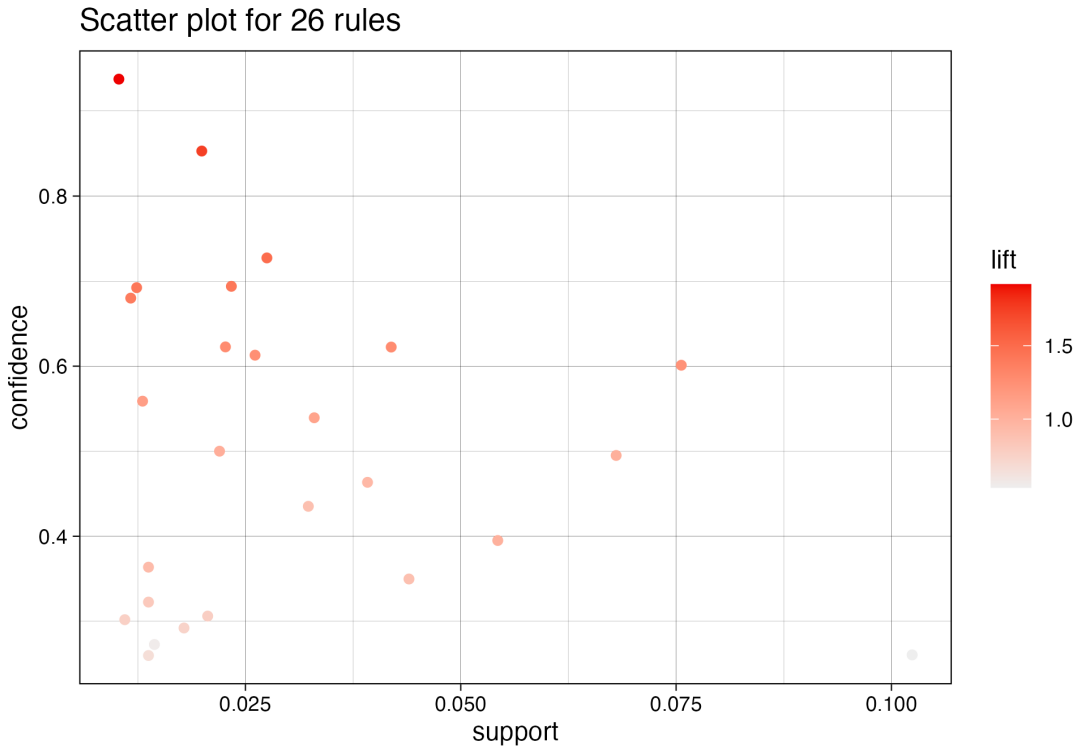
Tabla 5.4. Reglas encontradas con el algoritmo Apriori a las transacciones del fin de semana en la tarde según lift (cont.)

LHS		RHS	Soporte	Confianza	Cobertura	Lift
Pastry	=>	Bread	0.02	0.37	0.04	1.17
Cookies	=>	Coffee	0.02	0.54	0.05	1.16
Juice	=>	Coffee	0.03	0.54	0.05	1.16
Tiffin	=>	Coffee	0.01	0.53	0.02	1.14
Cake	=>	Coffee	0.08	0.52	0.16	1.13
Jammie Dodgers	=>	Coffee	0.01	0.52	0.03	1.13
Brownie	=>	Coffee	0.03	0.52	0.06	1.12
Alfajores	=>	Coffee	0.03	0.51	0.05	1.10
Scone	=>	Coffee	0.03	0.50	0.06	1.08
Frittata	=>	Coffee	0.01	0.47	0.03	1.02
Hot chocolate	=>	Coffee	0.03	0.47	0.07	1.01
Coffee, Hot chocolate	=>	Bread	0.01	0.31	0.03	0.97
Muffin	=>	Coffee	0.02	0.43	0.05	0.93
Alfajores	=>	Bread	0.02	0.29	0.05	0.92
Brownie	=>	Bread	0.02	0.28	0.06	0.88
Cookies	=>	Bread	0.01	0.28	0.05	0.86
Scone	=>	Bread	0.02	0.27	0.06	0.84
Tea	=>	Coffee	0.07	0.38	0.17	0.82
Muffin	=>	Bread	0.01	0.26	0.05	0.82
Hot chocolate	=>	Bread	0.02	0.25	0.07	0.79
Cake	=>	Bread	0.04	0.25	0.16	0.78
Bread	=>	Coffee	0.10	0.32	0.32	0.69

Fuente: elaboración propia.

En esta versión del libro en formato PDF este cuadro no es interactivo. Si deseas ver la versión interactiva que además contiene todas las reglas, puedes consultar la versión en formato HTML del libro (http://www.icesi.edu.co/editorial/intro-analisis-canastas-web/).

Figura 5.11. Visualización de las métricas de las reglas obtenidas al aplicar el algoritmo Apriori a las transacciones de los fines de semana en la tarde.



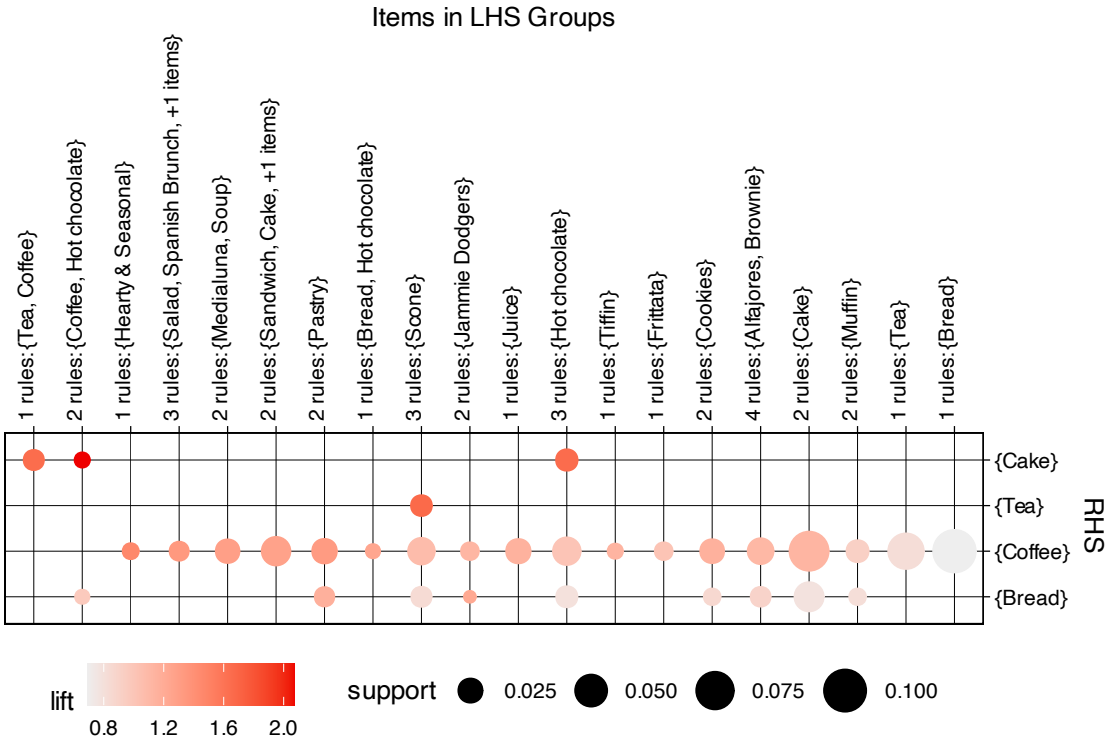
Fuente: elaboración propia.

En este formato PDF del libro la Figura 5.11 no es interactiva. Si deseas ver la versión interactiva, puedes consultar la versión en formato HTML del libro (http://www.icesi.edu.co/editorial/intro-analisis-canastas-web/).

La Figura 5.11 también permite observar claramente dos grupos: reglas con alto valor de *lift* pero bajo soporte (generalmente asociadas con pastel) y reglas con soporte medio y confianza alta, típicamente asociadas al café. La Figura 5.12 muestra que también tenemos reglas de asociación cuyo consecuente es el pan (*Bread*), pero siempre acompañado como consecuente de café o pastel.

Estos resultados subrayan diferencias notables en los patrones de consumo durante las tardes de los fines de semana respecto a los días laborales, destacándose especialmente por la centralidad del pastel y la aparición de combinaciones específicas propias del fin de semana.

Figura 5.12. Visualización de las reglas obtenidas al aplicar el algoritmo Apriori a las transacciones de los fines de semana en la tarde.



Fuente: elaboración propia.

5.6.3 Reglas para los días de la semana por la mañana

Las reglas identificadas por el científico de datos para los fines de semana por la tarde se reportan en el Cuadro 5.5. En las Figuras 5.13 y 5.14 se presentan visualizaciones de las métricas de las reglas.

El análisis realizado para las transacciones correspondientes a las mañanas de los días laborales identificó 17 reglas no redundantes con un soporte mínimo del 1% y una confianza mínima del 25%. En esta franja horaria, la mayoría de las reglas tienen como consecuente bebidas calientes, particularmente *Coffee*, aunque aparecen tres reglas de asociación con consecuente *Bread*. La regla más destacada en términos de *lift* es $\{\text{Keeping It Local}\} \rightarrow \{\text{Coffee}\}$, que registra un valor de 1.39. Esta regla sugiere que, durante las mañanas laborales, la compra de Keeping It Local incrementa en casi un 39% la probabilidad de adquirir café, respecto a lo que se esperaría si estas compras fueran independientes. De manera similar, otras combinaciones relevantes, como $\{Toast\} \rightarrow \{\text{Coffee}\}$ y $\{Alfajores\} \rightarrow \{\text{Coffee}\}$, presentan valores elevados de confianza (alrededor del 60%), pero cobertura relativamente baja, inferior al 10%.

Tabla 5.5. Reglas encontradas con el algoritmo Apriori a las transacciones de los días de la semana en la mañana según lift

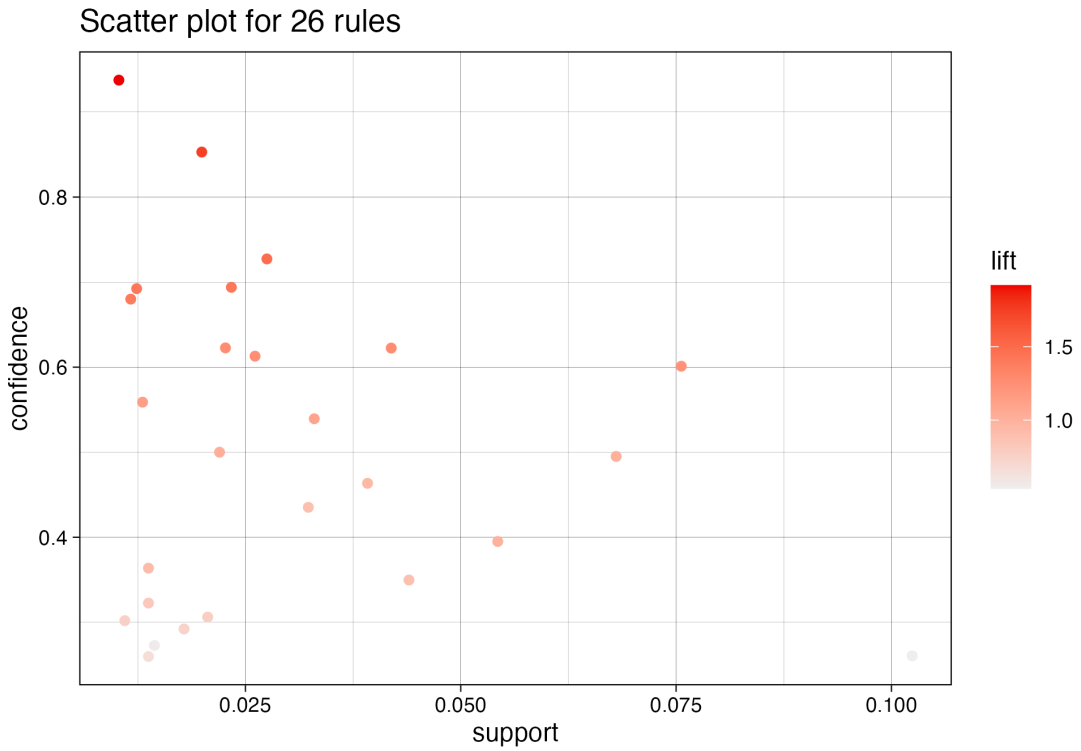
LHS		RHS	Soporte	Confianza	Cobertura	Lift
Keeping It Local	=>	Coffee	0.01	0.74	0.01	1.39
Toast	=>	Coffee	0.04	0.72	0.06	1.36
Alfajores	=>	Coffee	0.02	0.61	0.03	1.16
Cookies	=>	Coffee	0.03	0.59	0.05	1.12
Cake	=>	Coffee	0.03	0.59	0.06	1.11
Pastry	=>	Coffee	0.08	0.59	0.14	1.11
Medialuna	=>	Coffee	0.04	0.58	0.07	1.09
Muffin	=>	Coffee	0.02	0.58	0.03	1.09
Juice	=>	Coffee	0.02	0.56	0.03	1.06
Cookies	=>	Bread	0.02	0.33	0.05	0.95
Brownie	=>	Coffee	0.01	0.48	0.02	0.90
Hot chocolate	=>	Coffee	0.02	0.47	0.04	0.89
Pastry	=>	Bread	0.04	0.31	0.14	0.88
Medialuna	=>	Bread	0.02	0.28	0.07	0.82
Hot chocolate	=>	Bread	0.01	0.26	0.04	0.75
Tea	=>	Coffee	0.04	0.35	0.12	0.66
Bread	=>	Coffee	0.09	0.26	0.35	0.49

Fuente: elaboración propia.

En esta versión en formato PDF del libro este cuadro no es interactivo. Si deseas ver la versión interactiva que además contiene todas las reglas, puedes consultar la versión en formato HTML del libro (http://www.icesi.edu.co/editorial/intro-analisis-canastas-web/).

La Figura 5.13 muestra una concentración de reglas con soporte moderado (entre 2 % y 7 %) y confianza alta (superior al 50 %), especialmente aquellas cuyo consecuente es *Coffee*. Por otro lado, la Figura 5.14 proporciona una representación alternativa de estos resultados, en la que se visualiza claramente que las reglas con mayor soporte están asociadas al consumo de café. Esta figura complementa el análisis previo al mostrar visualmente la estructura y frecuencia de estas combinaciones en el segmento analizado.

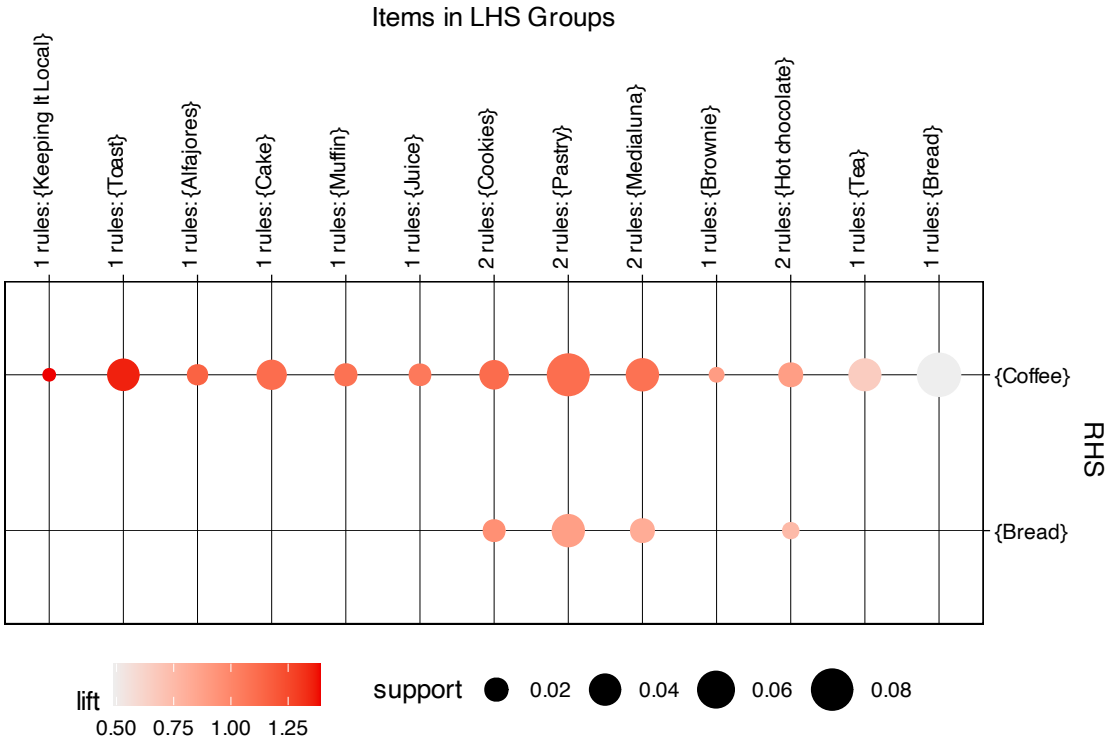
Figura 5.13. Visualización de las métricas de las reglas obtenidas al aplicar el algoritmo Apriori a las transacciones de los días de la semana en la mañana.



Fuente: elaboración propia.

En este formato PDF del libro la Figura 5.13 no es interactiva. Si deseas ver la versión interactiva, puedes consultar la versión en formato HTML del libro (http://www.icesi.edu.co/editorial/intro-analisis-canastas-web/).

Figura 5.14. Visualización de las reglas obtenidas al aplicar el algoritmo Apriori a las transacciones de los días de semana en la mañana.



Fuente: elaboración propia.

5.6.4 Reglas para fin de semana por la mañana

Las reglas identificadas por el científico de datos para los fines de semana por la tarde se reportan en el Cuadro 5.6. En las Figuras 5.15 y 5.16 se presentan visualizaciones de las métricas de las reglas. Para las transacciones de las mañanas de los fines de semana, el análisis identificó 26 reglas no redundantes con un soporte mínimo del 1% y una confianza mínima del 25%.

En esta franja sobresalen nuevamente reglas cuyo consecuente es el *Coffee*, acompañado como antecedente de productos típicos del desayuno o *brunch* como *Toast*, *Pastry* y Medialuna. La regla más destacada por su valor de *lift* es $\{\text{The Nomad}\} \rightarrow \{Coffee\}$, con un valor de 1.92. Es decir, durante las mañanas de fin de semana, la compra de $The Nomad $ incrementa en un 92% la probabilidad de comprar café, en comparación con la compra independiente de estos productos. Asimismo, la regla $\{\text{Spanish Brunch}\} \rightarrow \{Coffee\}$ presenta un alto valor de *lift* y confianza (85%).

Tabla 5.6. Reglas encontradas con el algoritmo Apriori a las transacciones de los fines de semana en la mañana según lift

LHS		RHS	Soporte	Confianza	Cobertura	Lift
The Nomad	=>	Coffee	0.01	0.94	0.01	1.92
Spanish Brunch	=>	Coffee	0.02	0.85	0.02	1.75
Toast	=>	Coffee	0.03	0.73	0.04	1.49
Juice	=>	Coffee	0.02	0.69	0.03	1.42
Sandwich	=>	Coffee	0.01	0.69	0.02	1.42
Tiffin	=>	Coffee	0.01	0.68	0.02	1.39
Cookies	=>	Coffee	0.02	0.62	0.04	1.27
Hot chocolate	=>	Coffee	0.04	0.62	0.07	1.27
Brownie	=>	Coffee	0.03	0.61	0.04	1.25
Medialuna	=>	Coffee	0.08	0.60	0.13	1.23
Alfajores	=>	Coffee	0.01	0.56	0.02	1.14
Scone	=>	Coffee	0.03	0.54	0.06	1.10
Muffin	=>	Coffee	0.02	0.50	0.04	1.02
Pastry	=>	Coffee	0.07	0.50	0.14	1.01
Pastry	=>	Bread	0.05	0.40	0.14	1.00
Tea	=>	Coffee	0.04	0.46	0.08	0.95
Toast	=>	Bread	0.01	0.36	0.04	0.92
Cake	=>	Coffee	0.03	0.44	0.07	0.89
Medialuna	=>	Bread	0.04	0.35	0.13	0.89
Brownie	=>	Bread	0.01	0.32	0.04	0.82
Hot chocolate	=>	Bread	0.02	0.31	0.07	0.78
Cookies	=>	Bread	0.01	0.30	0.04	0.77
Scone	=>	Bread	0.02	0.29	0.06	0.74
Farm House	=>	Bread	0.01	0.26	0.05	0.66
Farm House	=>	Coffee	0.01	0.27	0.05	0.56

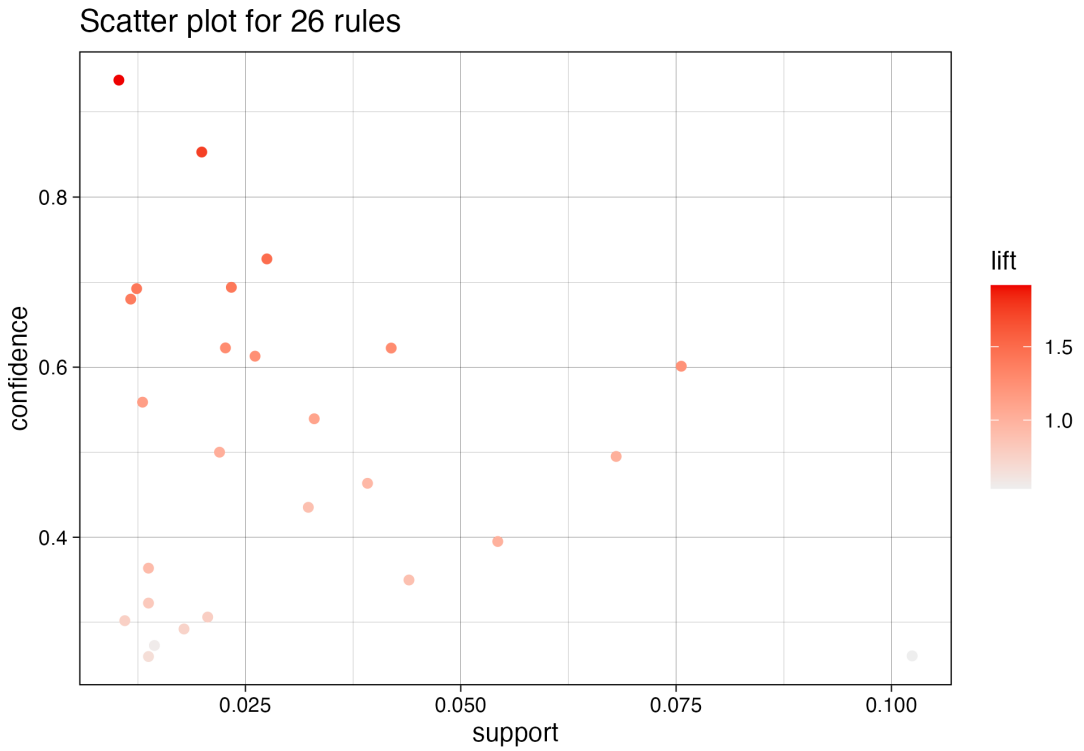
Tabla 5.6. Reglas encontradas con el algoritmo Apriori a las transacciones de los fines de semana en la mañana según lift (cont.)

LHS		RHS	Soporte	Confianza	Cobertura	Lift
Bread	=>	Coffee	0.10	0.26	0.39	0.53

Fuente: elaboración propia.

En este formato PDF del libro el Cuadro 5.6 no es interactivo. Si deseas ver la versión interactiva que además contiene todas las reglas, puedes consultar la versión en formato HTML del libro (http://www.icesi.edu.co/editorial/intro-analisis-canastas-web/).

Figura 5.15. Visualización de las métricas de las reglas obtenidas al aplicar el algoritmo Apriori a las transacciones del fin de semana en la mañana.



Fuente: elaboración propia.

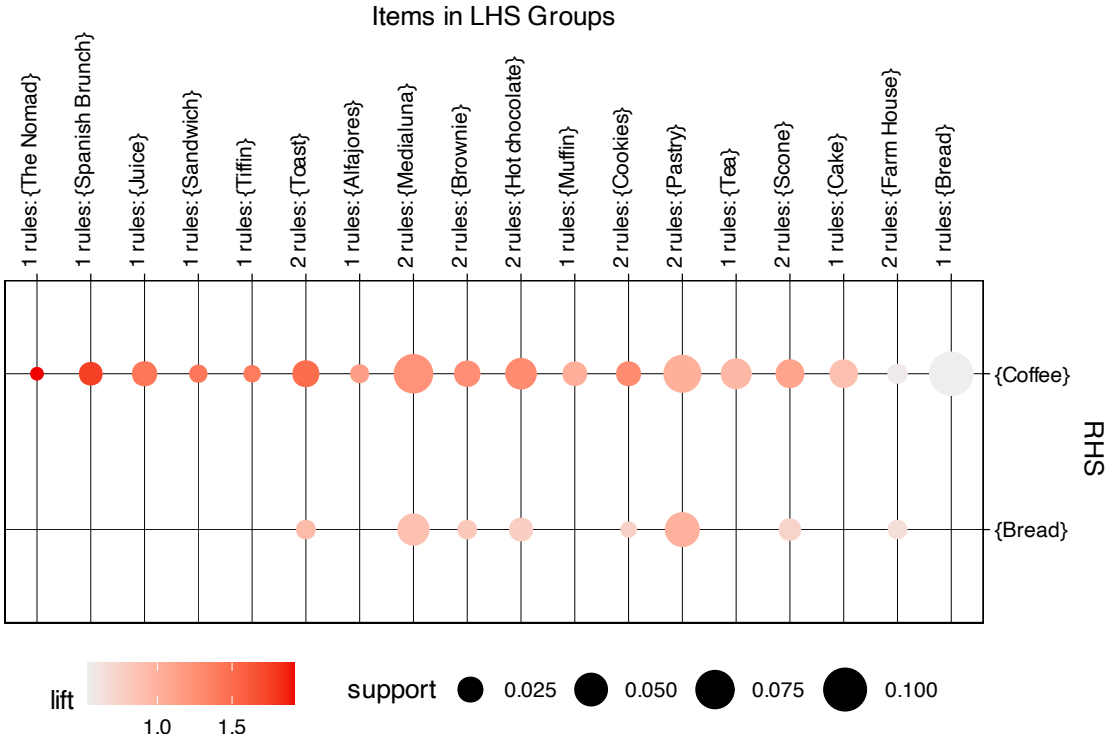
En esta en formato PDF del libro la Figura 5.15 no es interactiva. Si deseas ver la versión interactiva, puedes consultar la versión en formato HTML del libro (http://www.icesi.edu.co/editorial/intro-analisis-canastas-web/).

La Figura 5.15 muestra que la mayoría de las reglas se concentran alrededor de un

soporte del 2 % al 6 %. Adicionalmente, existen reglas menos frecuentes pero altamente relevantes desde la perspectiva del *lift*, lo que sugiere patrones muy específicos de consumo en estas mañanas de fin de semana.

Por su parte, la Figura 5.16 ofrece una visualización complementaria, evidenciando cómo las reglas con mayores valores de soporte involucran de manera consistente al café.

Figura 5.16. Visualización de las reglas obtenidas al aplicar el algoritmo Apriori a las transacciones de los fines de semana en la mañana.



Fuente: elaboración propia.

5.7 Insights

Con estos resultados, el científico de datos y el *analytics translator* se sentaron a generar los *insights* para el negocio. Para esto empezaron a responder las preguntas de negocios planteadas (Ver Sección 5.4). Ellos ya estaban convencidos de que la primera pregunta (*¿existe alguna diferencia entre las reglas de asociación según el tipo de día y el momento del día?*) ya se podía resolver. Tras examinar los cuatro subconjuntos resultantes de cruzar *weekday*/*weekend* con *morning*/*afternoon*, la respuesta era evidente.

En las **mañanas de los días de semana** se detectaron 17 reglas no redundantes: el 82 % culmina en *Coffee* y el 18 % en "Bread". Predomina un consumo funcional; reglas como $\{\text{Toast}\} \rightarrow \{Coffee\}$" o $\{\text{Farm House Toast}\} \rightarrow \{Coffee\}$ muestran *lifts* cercanos a 1.4 y confianzas en torno al 60 %, lo que sugiere que los consumidores están buscando energía rápida más que indulgencia para iniciar su jornada laboral. Cuando llega el **fin de semana por la mañana**, el número de reglas asciende a 26 y los consecuentes se reparten entre *Coffee* (69 %) y *Bread* (31 %). Aparece así un "desayuno relajado": combinaciones como $\{\text{Spanish Brunch}\} \rightarrow \{Coffee\}$" tienen soportes inferiores al 2 % pero *lifts* de 1.745, señal de un nicho que aprecia este tipo de desayunos. También aparecen reglas como $\{\text{Alfajores}\} \rightarrow \{Coffee\}$ que muestran algunos consumidores buscando un toque indulgente.

El panorama cambia en la **tarde de los días de semana**, donde emergen 21 reglas. *Coffee* concentra el 81 % de los consecuentes, seguido de *Tea* (9.5 %) y *Bread* (9.5 %). Las reglas $\{\text{Soupe}\} \rightarrow \{Tea\}$ y $\{\text{Salad}\} \rightarrow \{Coffee\}$ presentan un *lift* relativamente alto, mostrando un patrón de reglas más asociado a un almuerzo rápido. Finalmente, la **tarde de fin de semana** presenta 37 reglas. *Coffee* corresponde al 62 % de los consecuentes de las reglas encontradas, *Bread* al 27 % y *Cake* domina el 8 %. La regla $\{\text{Coffee, Hot chocolate}\} \rightarrow \{Cake\}$ presenta un *lift* superior a 2,0, reflejando un comprador que se premia con ítems dulces durante el fin de semana.

La comparación entre mañana y tarde confirma que el desayuno se centra en bebidas calientes y pan básico, mientras que las tardes añaden repostería y bebidas especiales, con *lifts* más elevados en sábado y domingo. Estos resultados muestran que *sí existe diferencia entre las reglas de asociación según el tipo de día y el momento del día*. Los resultados justifican la necesidad de estrategias diferenciadas de *bundling*, precios y exhibición según franja y tipo de día.

Tras responder la primera pregunta, el *analytics translator* y el científico de datos centran su atención a la tercera pregunta: *¿se puede crear un combo con sentido que sirva cualquier día y en cualquier momento del día?*

La primera reacción es responder no, pues las reglas son diferentes. No obstante, una lectura transversal de las reglas muestra que la canasta que contiene *Coffee* y *Pastry*, donde *Pastry* incluye medialunas, croissants y piezas de hojaldre, aparece con métricas consistentes en los cuatro escenarios analizados. En las mañanas de días laborales las reglas del tipo $\{\text{Pastry}\} \rightarrow \{Coffee\}$ alcanza un soporte del 8 % y una confianza del 59 %, con *lift* de 1.1; en las mañanas de fin de semana mantiene un soporte del 7 % y *lift* de 1.01; en las tardes laborales aparece con confianza del 64 % y *lift* de 1.41; y

en las tardes de fin de semana exhibe un soporte del 3 %, confianza del 61 % y *lift* de 1.32. Estas métricas indican que, aunque la intensidad varía, **la asociación se mantiene positiva y relevante en cualquier día y momento**.

Desde la lógica comercial, el combo *Coffee* y *Pastry* tiene sentido universal: responde al impulso de energía rápida de la mañana, al antojo de media tarde y al hábito indulgente del fin de semana. Su soporte agregado ronda el 6 % de todas las transacciones, suficiente para garantizar rotación, y el *lift* no cae por debajo de la unidad, lo que significa que el combo vende siempre más de lo esperado bajo independencia.

Por tanto, el *analytics translator* decidió llevar como propuesta construir un combo base: *Coffee* y *Pastry*, con un precio ligeramente inferior al ticket promedio, sin restricción horaria, permitiendo variar la pieza de pastelería según el inventario (croissant a primera hora, brownie por la tarde). De este modo se aprovecha una regla robusta y se ofrece al cliente una oferta coherente, fácil de comunicar y operativamente simple de implementar.

Cuando tuvieron claras las respuestas a estas dos preguntas, procedieron a generar *insights* para responder una pregunta que genera mucha controversia en la panadería: ¿tiene sentido hacer un descuento en el café? La respuesta a esta pregunta ya era evidente: **No tiene sentido hacer un descuento directo en el café**. Los resultados sugieren que es mejor concentrar los incentivos en los acompañamientos o en combos, no en la bebida que ya se vende sola.

Los resultados del caso muestran que *Coffee* es, por sí mismo, el producto de mayor frecuencia en todas las franjas y tipos de día, entre el 45 y el 65 % de las transacciones (Ver Figuras 5.8). Al mismo tiempo, las reglas donde el café actúa como consecuente apenas superan la independencia (*lift* de aproximadamente 1.1), lo que indica que las ventas del resto de los ítems apenas añaden probabilidad adicional sobre una demanda ya cautiva. Hacer un descuento en un artículo con alta rotación y bajo *lift* produciría una simple erosión del margen: los clientes seguirían pidiendo café, pagarían menos por él y el efecto arrastre sobre otros productos sería mínimo.

En cambio, las reglas con *lifts* realmente elevados (por ejemplo mayores a 1.5) involucran repostería o bebidas alternativas, lo que sugiere que **el incentivo debe concentrarse en el acompañamiento o en el *bundle*, no en la bebida ancla**. Por tanto, la estrategia óptima es mantener el precio del café y ofrecer promociones del tipo *Coffee* y *Pastry* con rebaja aplicada al pastel, o combos temáticos donde el té o el chocolate caliente se vendan con repostería premium. De esta forma se protege la rentabilidad del café, se impulsa el ticket medio y se potencian las asociaciones con mayor valor estratégico reveladas por el **Market Basket Analysis**.

Finalmente, el *analytics translator* y el científico de datos enfrentaron la última pregunta: ¿qué recomendaciones se pueden hacer para vender más té? Las reglas de asociación confirman que el té comparte un vínculo estructural con ítems "anzuelo" específicos, más que con descuentos directos en su propio precio. En la franja **tarde de los días de semana** la regla$\{\text{Soupe}\} \rightarrow \{Tea\}$ muestra el *lift* más alto (1.62) y una confianza cercana al 30 %; de forma análoga, en presencia de repostería premium (p. ej., *Brownie* o *Pastry*) el té mantiene *lifts* superiores a 1.4. Esto revela que la decisión

de añadir una infusión se activa cuando el cliente se orienta hacia productos salados ligeros o dulces indulgentes.

Empleando estos insumos, el *analytics translator* llega a la conclusión de que la promoción idónea no es rebajar el precio del té, sino empaquetarlo con los detonantes que elevan su probabilidad de compra. Un menú fijo "Sopa del día + Té", visible solo de 12:00 a 18:00 en días laborales, capitaliza la regla más poderosa y refuerza la percepción de almuerzo ligero. De modo complementario, la barra puede ofrecer un *upgrade* de bebida caliente a quienes eligen repostería premium: "¿Brownie? Pruébalo con nuestro té *Earl Grey*", dejando la rebaja (si la hay) en el pastel, no en la infusión. Para los sábados y domingos, donde la indulgencia se dispara, un paquete "Tea-time" de dos porciones de pastel y dos tés sueltos premium (stock limitado, 15–18 h) introduce sensación de ocasión especial y evita la canibalización del café.

El *analytics translator* también pensó en la visibilidad del producto. Los resultados soportan que puede ser una buena idea reubicar las variedades de té junto a sopas o ensaladas en la vitrina, y crear en la carta digital la sección "*Light lunch + Tea*", facilitan la asociación mental que refleja el MBA. Añadir un mensaje de valor —"Recarga de agua caliente gratis" entre las 10 h y las 12 h— prolonga la experiencia sin costo significativo y resulta especialmente atractivo para quienes trabajan desde la cafetería. Por último, propone una campaña personalizada en la app o por SMS, dirigida a clientes con reiteradas compras de repostería pero sin historial de té, puede ofrecer un cupón de 15 % para su primera infusión.

Con todos estos *insights*, el *analytics translator* y el científico de datos procedieron a construir una presentación que les permita transmitir todos los *insights* para facilitar la toma de decisiones del negocio.

5.8 Comentarios finales

A lo largo de este Capítulo seguimos el camino completo de un **MBA**, desde la depuración de la base del **Bread Basket** hasta la extracción, visualización e interpretación de reglas de asociación. Primero segmentamos los datos por tipo de día y franja horaria. Esto se debe a que los datos mostraban que los patrones de consumo y las canastas de compra no eran iguales. Por ejemplo, los clientes en la mañana laboral buscaban energía funcional, mientras que en la tarde del fin de semana se premiaban con repostería premium. Esta decisión metodológica, aunque parece simple en términos de cálculo, es muy importante para obtener información. Ayudó a encontrar señales que se habrían perdido en un análisis general, como el papel de la sopa para impulsar las ventas de té o la fuerza del café como producto principal. Como resultado, se encontró en un conjunto de reglas no redundantes, robustas y accionables, sustentadas con métricas de soporte, confianza y *lift*.

Los hallazgos tienen implicaciones estratégicas directas. Mantener el café sin promociones y centrar las ofertas en sus acompañantes preserva el margen de la bebida de mayor rotación. Los *bundles* "Coffee & Pastry" o "Sopa + Té" capturan patrones universales y de nicho sin necesidad de jugosos descuentos, mientras que la reubicación

física y digital de los tés junto a productos salados ligeros multiplica su visibilidad. Al mismo tiempo, el ejercicio de *business analytics* nos sugiere tácticas de personalización que van más allá del mostrador. Por ejemplo, un cupón dirigido a quienes compran brownies pero nunca té, o una notificación in-app que aparece cuando el carrito incluye sopa. Estas acciones convierten la minería de reglas en una palanca de CRM.

Desde un enfoque práctico, el caso muestra la importancia de establecer niveles de soporte y confianza que se alineen con la realidad del negocio. También destaca la necesidad de eliminar reglas repetitivas para no abrumar a los tomadores de decisiones con información innecesaria y elegir visualizaciones claras y útiles en lugar de complicadas. Las visualizaciones, bien combinadas, permiten al *analytics translator* conectar la intuición del gerente con la evidencia estadística del científico de datos.

Es importante reconocer las limitaciones del ejercicio. Trabajamos con un histórico de dieciocho meses y con tickets de un único punto de venta; ignoramos información de precios y márgenes unitarios; y no modelamos estacionalidad ni efectos de promociones pasadas. Cada una de estas ausencias es, al mismo tiempo, una oportunidad para expandir el análisis: reglas secuenciales, canastas en cliente, integración con pronósticos de demanda y pruebas A/B que midan el impacto financiero de los *bundles* propuestos.

Una implementación exitosa requerirá una hoja de ruta sencilla: ajustar combos, reexhibir productos clave, lanzar campañas personalizadas y refrescar el **MBA** de forma trimestral. El indicador final no será el número de reglas descubiertas, sino la evolución del ticket medio, la rotación de inventario y la satisfacción del cliente.

Con esta aplicación cerramos este libro. Mostrando cómo una técnica clásica, aplicada con rigor y traducida con claridad, se transforma en decisiones que generan valor tangible. Queda en tus manos experimentar, medir y compartir lo aprendido, alimentando un ciclo virtuoso de mejora continua. Recuerda que "en el mundo del *business analytics*, ¡la imaginación es el límite!".

Anexo con código del caso

El siguiente código fue empleado para obtener los resultados de esta Capítulo.

Carga y exploración de los datos

En esta sección se presenta el código empleado para generar los resultados de la Sección 5.3. Primero se cargan los paquetes y los datos.

```
# Cargar paquetes
library(tidyr)
library(dplyr)
library(lubridate)
library(data.table)
library(xts)
```

```
library(ggplot2)
library(forcats)


# Leer datos
pan <- read.csv("./05-Caso/bread basket.csv")
# Convertir period_day y weekday_weekend a factor
pan <- pan %>%
    mutate(across(c("period_day", "weekday_weekend"), as.factor))
```

El siguiente código permite construir el Cuadro 5.1.

```
# Descripción de los datos

tabla_period_day <- pan %>%
    select(-Item, -date_time) %>%
    unique() %>%
    select(-Transaction)

# Crear tabla de frecuencias (absolutas)
tabla_period_day <- table(tabla_period_day$period_day,
↪  tabla_period_day$weekday_weekend,
    dnn = c("Momento del día", "Tipo de día"))
tabla_period_day <- addmargins(tabla_period_day, margin = c(1, 2))
# Crear tabla de frecuencias (relativas)
tabla_period_day <- round(tabla_period_day/tabla_period_day[5, 3] * 100, 2)

tabla_period_day
```

Con el siguiente código se eliminan las observaciones para la noche.

```
# eliminar las observaciones que corresponden a night y evening

pan <- pan %>%
    filter(period_day %in% c("morning", "afternoon")) %>%
    droplevels()  # se requiere quitar los niveles de los factores que ya no
    ↪  son útiles
```

El Cuadro 5.2 se produce con las siguientes líneas de código:

```
tabla_period_day_2 <- pan %>%
    select(-Item, -date_time) %>%
    unique() %>%
    select(-Transaction)
```

```
tabla_period_day_2 <- table(tabla_period_day_2$period_day,
↪  tabla_period_day_2$weekday_weekend,
    dnn = c("Momento del día", "Tipo de día"))

tabla_period_day_2 <- addmargins(tabla_period_day_2, margin = c(1, 2))

tabla_period_day_2 <- round(tabla_period_day_2/tabla_period_day_2[3, 3] * 100,
↪  2)
```

El siguiente código genera el gráfico de calendario que se reporta en la Figura 5.1).

```
transacciones_todas <- full_join(transacciones_totales, transacciones_morning,
↪  by = "date") %>%
    full_join(transacciones_afternoon, by = "date")
# instalar el paquete si no se tiene install.packages('tbl2xts') Cargar el
# paquete

library(tbl2xts)

# Crear objeto de clase serie de tiempo (clase xts)
trans_total <- tbl_xts(transacciones_todas) %>%
    na.fill(fill = 0)
# Chequear la clase del objeto
class(trans_total)

# Guardar la variable date_time en formato fecha
temp <- as.POSIXlt(pan$date_time, format = "%d-%m-%Y %H:%M")

# instalar el paquete si no se tiene install.packages('lubridate') Cargar el
# paquete
library(lubridate)
# Crear la variable ´date´
pan$date <- date(temp)


# Crear la serie de tiempo de transacciones

transacciones_totales <- pan %>%
    select(Transaction, date) %>%
    group_by(date) %>%
    tally(name = "total")

transacciones_morning <- pan %>%
    filter(period_day == "morning") %>%
    select(Transaction, date) %>%
    group_by(date) %>%
```

```
    tally(name = "morning")

transacciones_afternoon <- pan %>%
    filter(period_day == "afternoon") %>%
    select(Transaction, date) %>%
    group_by(date) %>%
    tally(name = "afternoon")

transacciones_todas <- full_join(transacciones_totales, transacciones_morning,
↪  by = "date") %>%
    full_join(transacciones_afternoon, by = "date")
# instalar el paquete si no se tiene install.packages('tbl2xts') Cargar el
# paquete

library(tbl2xts)

# Crear objeto de clase serie de tiempo (clase xts)
trans_total <- tbl_xts(transacciones_todas) %>%
    na.fill(fill = 0)
# Chequear la clase del objeto
class(trans_total)

# instalar el paquete si no se tiene install.packages('openair') Cargar el
# paquete
library(openair)

# instalar el paquete si no se tiene install.packages('RColorBrewer') Cargar
↪  el
# paquete

library(RColorBrewer)

# crear paleta de 9 colores azules
colores <- brewer.pal(9, "PuBu")

calendarPlot(transacciones_todas, pollutant = "total", year = c(2016, 2017),
↪  key.position = "bottom",
    cols = colores)
```

Las Figura 5.4 y la Figura 5.5 fueron construidas con el siguiente código:

```
transacciones_todas$day <- weekdays(transacciones_todas$date)

transacciones_todas %>%
    pivot_longer(c("total", "morning", "afternoon"), names_to = "Momento",
    ↪  values_to = "Transacciones") %>%
```

```
    mutate_at("Momento", as.factor) %>%
    mutate(weekday_weekend = ifelse(day %in% c("Saturday", "Sunday"),
    ↪  "weekend",
        "weekday")) %>%
    filter(Momento != "total") %>%
    ggplot(aes(x = weekday_weekend, y = Transacciones, col = Momento)) +
    ↪  scale_color_brewer(palette = "Dark2",
    type = "div") + scale_fill_brewer(palette = "Dark2", type = "div") +
    ↪  geom_boxplot(fill = "white",
    outlier.shape = NA) + geom_jitter(size = 0.85, alpha = 0.5) + ylab("Tipo de
    ↪  día") +
    coord_flip() + theme_minimal() + theme(panel.grid.major = element_blank(),
    ↪  panel.grid.minor = element_blank(),
    legend.position = "bottom", axis.line = element_line(linewidth = 0.5,
    ↪  colour = "gray"))


transacciones_todas %>%
    pivot_longer(c("total", "morning", "afternoon"), names_to = "Momento",
    ↪  values_to = "Transacciones") %>%
    mutate_at("Momento", as.factor) %>%
    mutate(weekday_weekend = ifelse(day %in% c("Saturday", "Sunday"),
    ↪  "weekend",
        "weekday")) %>%
    filter(Momento != "total") %>%
    ggplot(aes(col = weekday_weekend, y = Transacciones, x = Momento)) +
    ↪  geom_boxplot(fill = "white",
    outlier.shape = NA) + geom_jitter(size = 0.85, alpha = 0.5) + coord_flip()
    ↪  +
    theme_minimal() + theme(panel.grid.major = element_blank(),
    ↪  panel.grid.minor = element_blank(),
    legend.position = "bottom", axis.text.x = element_text(size = 4.5, angle =
    ↪  0),
    axis.text.y = element_text(size = 5), axis.line = element_line(linewidth =
    ↪  0.5,
        colour = "gray"))
```

Las Figuras 5.6 y 5.7 fueron construidas con el siguiente código:

```
transacciones_todas %>%
    pivot_longer(c("total", "morning", "afternoon"), names_to = "Momento",
    ↪  values_to = "Transacciones") %>%
    mutate_at("Momento", as.factor) %>%
    mutate(weekday_weekend = ifelse(day %in% c("Saturday", "Sunday"),
    ↪  "weekend",
        "weekday")) %>%
```

```
    filter(Momento != "total") %>%
    ggplot(aes(col = Momento, x = Transacciones, fill = Momento)) +
    ↪  scale_color_brewer(palette = "Dark2",
    type = "div") + scale_fill_brewer(palette = "Dark2", type = "div") +
    ↪  geom_density(alpha = 0.6) +
    ylab("Densidad") + facet_wrap(~weekday_weekend) + theme_minimal() +
    ↪  theme(panel.grid.major = element_blank(),
    panel.grid.minor = element_blank(), legend.position = "bottom", axis.line =
    ↪  element_line(linewidth = 0.5,
        colour = "gray"))

transacciones_todas %>%
    pivot_longer(c("total", "morning", "afternoon"), names_to = "Momento",
    ↪  values_to = "Transacciones") %>%
    mutate_at("Momento", as.factor) %>%
    mutate(weekday_weekend = ifelse(day %in% c("Saturday", "Sunday"),
    ↪  "weekend",
        "weekday")) %>%
    filter(Momento != "total") %>%
    ggplot(aes(col = weekday_weekend, x = Transacciones, fill =
    ↪  weekday_weekend)) +
    geom_density(alpha = 0.6) + ylab("Densidad") + facet_wrap(~Momento) +
    ↪  theme_minimal() +
    theme(panel.grid.major = element_blank(), panel.grid.minor =
    ↪  element_blank(),
        legend.position = "bottom", axis.line = element_line(linewidth = 0.5,
        ↪  colour = "gray"))
```

La Figura 5.8 fue creada con el siguiente código:

```
## creación de datos

# datos afternoon weekday

datos_aft_week <- pan %>%
    filter(period_day == "afternoon") %>%
    filter(weekday_weekend == "weekday")
# número de transacciones en afternoon weekday
num_transacciones_aft_week <- n_distinct(datos_aft_week$Transaction)

top_productos_aft_week <- datos_aft_week %>%
    group_by(Item) %>%
    summarise(proporcion = n()/num_transacciones_aft_week) %>%
    top_n(10, proporcion) %>%
    arrange(proporcion)
```

```r
# datos afternoon weekday

datos_aft_weekend <- pan %>%
    filter(period_day == "afternoon") %>%
    filter(weekday_weekend == "weekend")

# número de transacciones en afternoon weekday
num_transacciones_aft_weekend <- n_distinct(datos_aft_weekend$Transaction)

top_productos_aft_weekend <- datos_aft_weekend %>%
    group_by(Item) %>%
    summarise(proporcion = n()/num_transacciones_aft_weekend) %>%
    top_n(10, proporcion) %>%
    arrange(proporcion)

# datos morning weekday

datos_mor_week <- pan %>%
    filter(period_day == "morning") %>%
    filter(weekday_weekend == "weekday")

# número de transacciones en morning weekday
num_transacciones_mor_week <- n_distinct(datos_mor_week$Transaction)

top_productos_mor_week <- datos_mor_week %>%
    group_by(Item) %>%
    summarise(proporcion = n()/num_transacciones_mor_week) %>%
    top_n(10, proporcion) %>%
    arrange(proporcion)

# datos morning weekend

datos_mor_weekend <- pan %>%
    filter(period_day == "morning") %>%
    filter(weekday_weekend == "weekend")
# número de transacciones en morning weekend
num_transacciones_mor_weekend <- n_distinct(datos_mor_weekend$Transaction)

top_productos_mor_weekend <- datos_mor_week %>%
    group_by(Item) %>%
    summarise(proporcion = n()/num_transacciones_mor_weekend) %>%
    top_n(10, proporcion) %>%
    arrange(proporcion)
```

```
library(gghighlight)
p1 <- top_productos_mor_week %>%
    mutate(Item = factor(Item, Item)) %>%
    ggplot(aes(x = Item, y = proporcion)) + geom_col(fill = blues9[6]) +
    ↪  coord_flip()
p1 <- p1 + labs(y = "Frecuencia relativa (proporción 0-1)", x = "item", title =
↪  "Mañanas en días de la semana") +
    theme_minimal() + gghighlight(Item %in% c("Coffee", "Bread", "Pastry"),
    ↪  use_direct_label = F)




p2 <- top_productos_aft_week %>%
    mutate(Item = factor(Item, Item)) %>%
    ggplot(aes(x = Item, y = proporcion)) + geom_col(fill = blues9[7]) +
    ↪  coord_flip()
p2 <- p2 + labs(y = "Frecuencia relativa (proporción 0-1)", x = "item", title =
↪  "Tardes en días de la semana") +
    theme_minimal() + gghighlight(Item %in% c("Coffee", "Bread", "Tea"),
    ↪  use_direct_label = F)


p3 <- top_productos_mor_weekend %>%
    mutate(Item = factor(Item, Item)) %>%
    ggplot(aes(x = Item, y = proporcion)) + geom_col(fill = blues9[6]) +
    ↪  coord_flip()
p3 <- p3 + labs(y = "Frecuencia relativa (proporción 0-1)", x = "item", title =
↪  "Mañanas en fin de semana") +
    theme_minimal() + gghighlight(Item %in% c("Coffee", "Bread", "Pastry"),
    ↪  use_direct_label = F)


p4 <- top_productos_aft_weekend %>%
    mutate(Item = factor(Item, Item)) %>%
    ggplot(aes(x = Item, y = proporcion)) + geom_col(fill = blues9[7]) +
    ↪  coord_flip()
p4 <- p4 + labs(y = "Frecuencia relativa (proporción 0-1)", x = "item", title =
↪  "Tardes en fin de semana") +
    theme_minimal() + gghighlight(Item %in% c("Coffee", "Bread", "Tea"),
    ↪  use_direct_label = F)

library(ggpubr)
```

```
figura <- ggarrange(p1, p2, p3, p4, ncol = 2, nrow = 2)

figura
```

5.8.1 Modelado

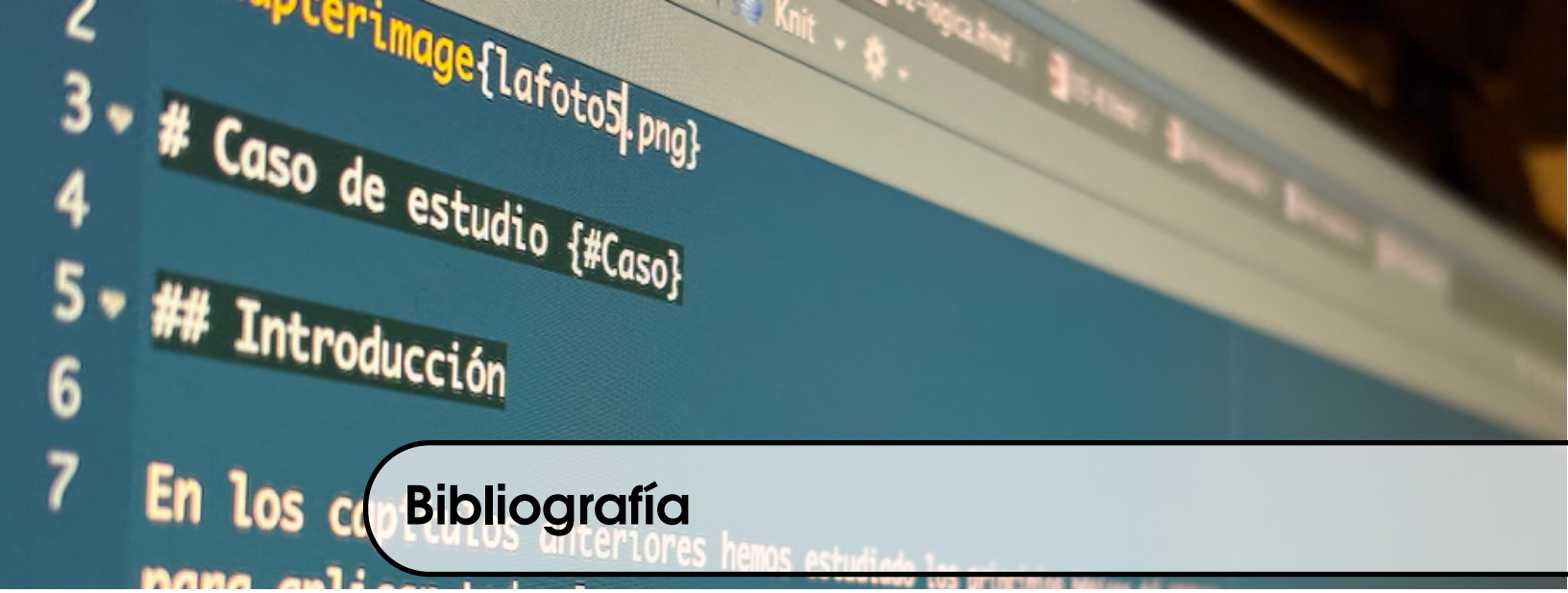
En esta sección se presenta el código que generó los resultados de las secciones 5.5 y 5.6. Las reglas de asociación se encuentran con el siguiente código:

```
library(arules)
## creación de datos

# datos afternoon weekday

datos_aft_week <- pan %>%
    filter(period_day == "afternoon") %>%
    filter(weekday_weekend == "weekday")

datos_aft_week <- split(datos_aft_week$Item, datos_aft_week$Transaction)

datos_tra_aft_week <- as(datos_aft_week, "transactions")

# datos afternoon weekday

datos_aft_weekend <- pan %>%
    filter(period_day == "afternoon") %>%
    filter(weekday_weekend == "weekend")

datos_aft_weekend <- split(datos_aft_weekend$Item,
↪  datos_aft_weekend$Transaction)

datos_tra_aft_weekend <- as(datos_aft_weekend, "transactions")

# datos morning weekday

datos_mor_week <- pan %>%
    filter(period_day == "morning") %>%
    filter(weekday_weekend == "weekday")

datos_mor_week <- split(datos_mor_week$Item, datos_mor_week$Transaction)

datos_tra_mor_week <- as(datos_mor_week, "transactions")

# datos morning weekday

datos_mor_weekend <- pan %>%
```

```
    filter(period_day == "morning") %>%
    filter(weekday_weekend == "weekend")

datos_mor_weekend <- split(datos_mor_weekend$Item,
↪  datos_mor_weekend$Transaction)

datos_tra_mor_weekend <- as(datos_mor_weekend, "transactions")
```

La visualización interactiva de las reglas se generan con el siguiente código:

```
# Tabla con resultados con solo 3 decimales

quality(reglas_aft_week) <- round(quality(reglas_aft_week), digits = 3)
inspectDT(reglas_aft_week)


plot(reglas_aft_week, engine = "plotly")

library(arulesViz)
# analissi gráfico de los resultados
plot(reglas_aft_week, method = "grouped")
```



Bibliografía

(2015). Online Retail. UCI Machine Learning Repository. DOI: https://doi.org/10.24432/C5BW33.

Agrawal, R., Srikant, R., et al. (1994). Fast algorithms for mining association rules. In *Proc. 20th int. conf. very large data bases, VLDB*, volume 1215, pages 487–499. Citeseer.

Al-Monawer, N., Davoodi, M., y Qi, L. (2021). Brand and quality effects on introduction of store brand products. *Journal of Retailing and Consumer Services*, 61:102507.

Alonso, J. C. (2021). Una introducción a los loops en r (y algunas alternativas). Technical report, Universidad Icesi.

Alonso, J. C. (2022). *Empezando a transformar bases de datos con R y dplyr*. Universidad Icesi.

Alonso, J. C. (2024). *Introducción al Modelo Clásico de Regresión para Científico de Datos en R*. Universidad Icesi.

Alonso, J. C. y Carabali, J. A. (2019). Breve tutorial para visualizar y calcular métricas de redes (grafos) en r (para económisas). Technical report, Universidad Icesi.

Alonso, J. C. y Largo, M. F. (2023). *Empezando a visualizar datos con R y ggplot2*. Universidad Icesi, 2. edition.

Alonso, J. C. y Ocampo, M. P. (2022). *Empezando a usaR: Una guía paso a paso*. Universidad Icesi.

Arboleda, A. M. y Alonso, J. C. (2016). Estimación de un modelo econométrico para determinar el efecto de acciones de marketing en ventas de productos de cuidado personal en Colombia. *Revista de Métodos Cuantitativos para la Economía y la Empresa*, 22:Páginas 230 a 249.

Arboleda, A. M. y Arce-Lopera, C. (2015). Quantitative analysis of product categorization in soft drinks using bottle silhouettes. *Food Quality and Preference*, 45:1–10.

CBR, S. W. (1998). Urban myth disproved: Beer and diapers don't mix.

Chang, W., Cheng, J., Allaire, J., Sievert, C., Schloerke, B., Xie, Y., Allen, J., McPherson, J., Dipert, A., y Borges, B. (2021). *shiny: Web Application Framework for R*. R package version 1.7.1.

Choi, P. (2018). Why do certain products influence grocery store choice? the role of anchor products and their relationships with other store choice factors: An abstract. In Krey, N. y Rossi, P., editors, *Back to the Future: Using Marketing Basics to Provide Customer Value*, pages 249–249, Cham. Springer International Publishing.

Contemporary Analysis, C. (2022). Diapers, beer, and data science in retail.

Csardi, G. y Nepusz, T. (2006). The igraph software package for complex network research. *InterJournal*, Complex Systems:1695.

Drèze, X. y Hoch, S. J. (1998). Exploiting the installed base using cross-merchandising and category destination programs. *International Journal of Research in Marketing*, 15(5):459–471.

Egbeola, S. (2023). The bread basket bakery — analysis project.

Hahsler, M. (2017). arulesViz: Interactive visualization of association rules with R. *R Journal*, 9(2):163–175.

Hahsler, M., Chelluboina, S., Hornik, K., y Buchta, C. (2011). The arules r-package ecosystem: Analyzing interesting patterns from large transaction datasets. *Journal of Machine Learning Research*, 12:1977–1981.

Hery, H. y Widjaja, A. E. (2024). Analysis of apriori and fp-growth algorithms for market basket insights: A case study of the bread basket bakery sales. *Journal of Digital Marketing and Digital Currency*, 1(1):63–74.

Kocas, C., Pauwels, K., y Bohlmann, J. D. (2018). Pricing best sellers and traffic generators: The role of asymmetric cross-selling. *Journal of Interactive Marketing*, 41(1):28–43.

Kotler, P. y Armstrong, G. (2012). *Marketing*. Pearson Educación, México, 10 edition.

lukeA (2017). Item frequency plots from object of class transactions in ggplot2.

Madsen, M. (2017). Beer, dispers and correlation: A tale of ambiguity.

Mittal, V. (2018). The Bread Basket. Dataset.

Oliveira, A. (2018). Bakery (market basket analysis). Kaggle Notebook.

Pascucci, F., Nardi, L., Marinelli, L., Paolanti, M., Frontoni, E., y Gregori, G. L. (2022). Combining sell-out data with shopper behaviour data for category performance measurement: The role of category conversion power. *Journal of Retailing and Consumer Services*, 65:102880.

Power, D. J. (2002). Ask dan!

R Core Team (2023). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria.

Rooderkerk, R. P. y Lehmann, D. R. (2021). Incorporating consumer product categorizations into shelf layout design. *Journal of Marketing Research*, 58(1):50–73.

Sievert, C. (2020). *Interactive Web-Based Data iczation with R, plotly, and shiny*. Chapman and Hall/CRC.

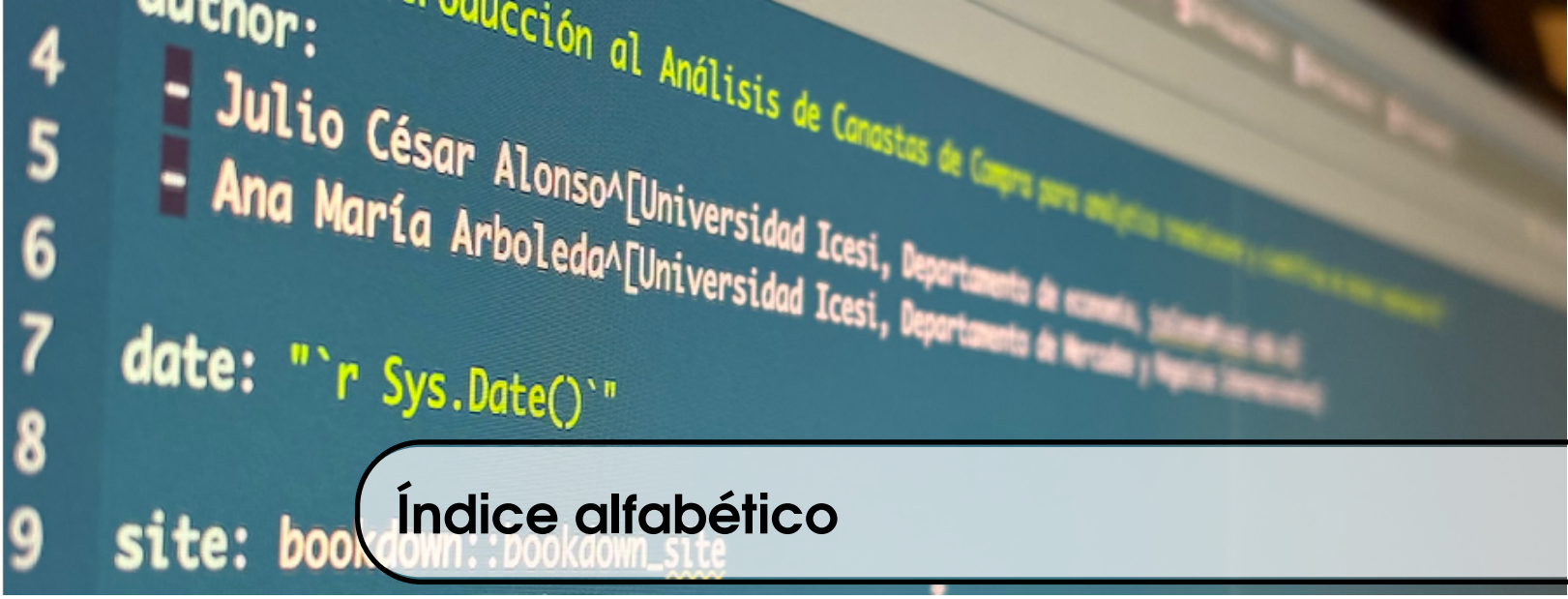
Swoyer, S. (2016). Beer and diapers: The impossible correlation.

Vaidyanathan, R., Xie, Y., Allaire, J., Cheng, J., Sievert, C., y Russell, K. (2022). *htmlwidgets: HTML Widgets for R*. R package version 1.5.4.

Wickham, H. (2016). *ggplot2: Elegant Graphics for Data Analysis*. Springer-Verlag New York.

Wickham, H., Averick, M., Bryan, J., Chang, W., McGowan, L. D., François, R., Grolemund, G., Hayes, A., Henry, L., Hester, J., Kuhn, M., Pedersen, T. L., Miller, E., Bache, S. M., Müller, K., Ooms, J., Robinson, D., Seidel, D. P., Spinu, V., Takahashi, K., Vaughan, D., Wilke, C., Woo, K., y Yutani, H. (2019). Welcome to the tidyverse. *Journal of Open Source Software*, 4(43):1686.

Wickham, H., François, R., Henry, L., y Müller, K. (2021). *dplyr: A Grammar of Data Manipulation*. R package version 1.0.7.



Índice alfabético

Universidad Icesi | Editorial Universidad Icesi

El análisis de canastas o de cesta de compra (en inglés es conocido como *Market Basket Analysis* o simplemente por la sigla MBA) es una herramienta poderosa en el mercadeo. Permite entender mejor el comportamiento y los hábitos de compra de los clientes cuando se cuenta con datos transaccionales. En especial, el MBA encuentra reglas de asociación que permiten identificar qué productos suelen comprarse juntos. Como se discutirá en esta obra, las reglas de asociación son útiles, por ejemplo, para desarrollar estrategias de ventas cruzadas y promociones personalizadas. Este libro está dirigido a dos roles en el mundo del business analytics: el científico de datos y el *analytics translator*.

Herramientas del BIG DATA y analytics